



**Carlos Filipe Abrantes Calmeiro**

Licenciado em Ciências da Engenharia Electrotécnica e de  
Computadores

**Tolerância a falhas com base em comutação  
de controladores – Implementação em  
autómatos programáveis**

Dissertação para obtenção do Grau de Mestre em Engenharia  
Electrotécnica e de Computadores

Orientador: Doutor Luís Filipe Figueira Brito Palma,  
Professor Auxiliar, Faculdade de Ciências e Tecnologia  
da Universidade Nova de Lisboa

Júri:

Presidente: Doutor Luís Filipe dos Santos Gomes  
Arguente: Doutor João Almeida das Rosas  
Vogal: Doutor Luís Filipe Figueira de Brito Palma





**Carlos Filipe Abrantes Calmeiro**

Licenciado em Ciências da Engenharia Electrotécnica e de  
Computadores

**Tolerância a falhas com base em comutação  
de controladores – Implementação em  
autómatos programáveis**

Dissertação para obtenção do Grau de Mestre em Engenharia  
Electrotécnica e de Computadores

Orientador: Doutor Luís Filipe Figueira Brito Palma,  
Professor Auxiliar, Faculdade de Ciências e Tecnologia  
da Universidade Nova de Lisboa

Júri:

Presidente: Doutor Luís Filipe dos Santos Gomes  
Arguente: Doutor João Almeida das Rosas  
Vogal: Doutor Luís Filipe Figueira de Brito Palma





# Copyright

**Tolerância a Falhas com base em Comutação de Controladores - Implementação em Autómatos Programáveis.**

Copyright ©2013 Carlos Filipe Abrantes Calmeiro; Faculdade de Ciências e Tecnologia; Universidade Nova de Lisboa.

A Faculdade de Ciências e Tecnologia e a Universidade Nova de Lisboa têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.



# Agradecimentos

Ao Professor Doutor Luís Brito Palma, agradeço o apoio e compreensão que me dispensou, a criação de condições para o desenvolvimento deste trabalho e o constante incentivo e entusiasmo ao longo desta jornada.

Ao amigo Carlos Carvalho, pela amizade, companheirismo e partilha de conhecimentos, não só durante a elaboração deste projecto, como também ao longo de todos estes anos que nos conhecemos.

Aos meus amigos e colegas de curso, em particular aos colegas Bruno Valente, Fábio Alves, Fábio Júlio, João Santos, Luís Lopes, Luís Miranda, Micael Simões, Miguel Marques, Pedro Gomes, Ricardo Mendonça e Tiago Xavier, o meu agradecimento pela sua amizade e companheirismo.

Ao terminar este trabalho não poderia deixar de agradecer a toda a minha família por tudo o que me têm dado e pela forma como têm contribuído para a minha formação.

Aos meus amigos André Palma, David Nunes, Diogo Estevão, Hugo Mariani, João Madureira, Mauro Guerreiro, Miguel Ribeiro, Pedro Mauricio, Pedro Leonardo, Ricardo Monteiro e Tiago Jacob, pela amizade demonstrada durante os últimos 9 anos.

Por fim, quero agradecer especialmente à minha Mãe e Avós.



# Resumo

Esta Tese resulta do trabalho realizado na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, sendo o principal objectivo o desenvolvimento de um sistema de controlo tolerante a falhas. O projecto desenvolvido aborda algumas das metodologias já existentes para o desenvolvimento deste tipo de sistemas e apresenta algumas propostas para a sua implementação em autómatos programáveis.

O sistema de controlo proposto tem por base a comutação de controladores, enquadrando-se por isso nos métodos activos de controlo tolerante a falhas. A abordagem proposta considera a existência de um sistema complementar de detecção, isolamento e identificação de falhas, fora do âmbito tese.

De forma a garantir a independência entre o projecto de controlo e o de detecção e diagnóstico de falhas, é proposto que a implementação se realize em autómatos distintos, acrescentando ao projecto uma componente de comunicação entre dispositivos. A implementação proposta para as estruturas de controlo tem por base a utilização das linguagens de programação descritas na norma 61311-3 do IEC (*Internacional Electrotechnical Commission*) - Texto estruturado e Diagrama funcional de blocos.

Para concluir sobre a aplicabilidade e desempenho das propostas sugeridas nesta tese, são apresentados os resultados do sistema de controlo quando aplicado ao processo laboratorial de dois tanques FBK 38-100, instalado no laboratório de automação da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa.

**Palavras-chave:** Sistemas dinâmicos, Controlo Tolerante a Falhas; Autómatos programáveis.



# Abstract

This thesis results from the work realized in Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa with the main objective of developing a fault tolerant control system. The project covers some of the existing methodologies for the development of such systems and presents some proposals for its implementation on a PLC (Programmable Logic Controller).

The proposed control system is based on switching controllers and it fits in the active fault tolerant control methods. The proposed approach considers the existence of a complementary system of detection, isolation and identification of faults, which is outside the scope of thesis.

To ensure the independence between the Control and Faults Detection projects is proposed the implementation in different PLCs, adding a communication component to the project. The proposed implementation for control structures is based on the use of programming languages described in the IEC (International Electrotechnical Commission) standard 61311-3 - Structured Text and Functional Block Diagram.

To conclude on the applicability and performance of the proposals suggested in this thesis were presented the results of the control system when applied to the laboratorial process of two tanks FBK 38-100 installed in the automation laboratory in Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa.

**Keywords:** Dynamic Systems, Fault Tolerant Control; PLCs.





# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Introdução . . . . .	2
1.2	Motivações . . . . .	3
1.3	Objectivos e Contribuições . . . . .	3
1.4	O Sistema Proposto de Controlo Tolerante a Falhas . . . . .	3
1.5	Organização da Tese . . . . .	4
<b>2</b>	<b>Estado de Arte</b>	<b>5</b>
2.1	Controlo Tolerante a Falhas . . . . .	6
2.1.1	Métodos Passivos de Controlo Tolerante a Falhas . . . . .	6
2.1.2	Métodos Activos de Controlo Tolerante a Falhas . . . . .	7
2.2	Autómatos Programáveis . . . . .	11
2.2.1	Diagrama em Escada . . . . .	13
2.2.2	Lista de Instruções . . . . .	13
2.2.3	Grafo Funcional Sequencial . . . . .	14
2.2.4	Texto Estruturado . . . . .	14
2.2.5	Diagrama Funcional de Blocos . . . . .	15
<b>3</b>	<b>Sistema de Controlo Proposto</b>	<b>17</b>
3.1	Arquitectura . . . . .	18
3.1.1	Nível de Processo . . . . .	19
3.1.2	Nível de Controlo . . . . .	20
3.1.3	Nível de Supervisão . . . . .	21
3.2	Metodologia . . . . .	22

3.2.1	Decisor . . . . .	22
3.2.2	Sensores e Actuadores Virtuais . . . . .	24
3.2.3	Controlador . . . . .	25
<b>4</b>	<b>Resultados Experimentais</b>	<b>31</b>
4.1	Arquitectura . . . . .	32
4.1.1	Nível de Processo . . . . .	33
4.1.2	Nível de Controlo . . . . .	36
4.1.3	Nível de Supervisão . . . . .	36
4.2	Metodologia . . . . .	37
4.3	Implementação . . . . .	41
4.3.1	Configuração de <i>Hardware</i> . . . . .	41
4.3.2	Configuração de rede comunicação . . . . .	41
4.3.3	Definição de variáveis . . . . .	42
4.3.4	Aquisição de dados . . . . .	45
4.3.5	Comunicação . . . . .	45
4.3.6	Estrutura de controlo . . . . .	47
4.3.7	Actuação . . . . .	48
4.4	Resultados . . . . .	48
4.4.1	Ensaio em Funcionamento Nominal . . . . .	48
4.4.2	Ensaaios com Falhas . . . . .	49
<b>5</b>	<b>Conclusão</b>	<b>53</b>
5.1	Conclusões . . . . .	54
5.2	Trabalho Futuro . . . . .	55

# Lista de Figuras

2.1	Controlo Tolerante a Falhas - Principais Metodologias . . . . .	6
2.2	Métodos Activos de Controlo Tolerante a Falhas - Arquitectura geral . . . . .	8
2.3	Método de Múltiplos Modelos - Arquitectura . . . . .	9
2.4	Método de Comutação de Controladores - Arquitectura . . . . .	9
2.5	Método de Controlo Preditivo - Arquitectura . . . . .	10
2.6	Método de Controlo Adaptativo - abordagem directa. . . . .	11
2.7	Método de Controlo Adaptativo - abordagem indirecta. . . . .	11
2.8	Autómatos Programáveis - Arquitectura . . . . .	12
2.9	Diagrama de Escada - Exemplo . . . . .	13
2.10	Lista de Instruções - Exemplo . . . . .	13
2.11	Grafo Funcional Sequencial - Exemplo . . . . .	14
2.12	Texto Estruturado - Exemplo 1. . . . .	14
2.13	Texto Estruturado - Exemplo 2. . . . .	14
2.14	Texto Estruturado - Exemplo 3. . . . .	15
2.15	Texto Estruturado - Exemplo 4. . . . .	15
2.16	Diagrama Funcional de Blocos - Exemplo . . . . .	15
3.1	Arquitectura de controlo proposta. . . . .	18
3.2	Nível de Processo. . . . .	19
3.3	Nível de Controlo . . . . .	20
3.4	Nível de Supervisão. . . . .	21
3.5	Decisor . . . . .	22
3.6	Decisor - Implementação em ST . . . . .	23
3.7	Decisor - Implementação em FBD . . . . .	24

3.8	Actuador Virtual . . . . .	24
3.9	Sensor Virtual . . . . .	24
3.10	Actuador Virtual - Implementação em ST . . . . .	25
3.11	Sensor Virtual - Implementação em ST . . . . .	25
3.12	Actuador Virtual - Implementação em FBD . . . . .	25
3.13	Sensor Virtual - Implementação em FBD . . . . .	25
3.14	Controlador. . . . .	25
3.15	Exemplo de <i>Windup</i> . . . . .	27
3.16	<i>Anti-Windup</i> - Implementação em ST . . . . .	28
3.17	Comutação - Implementação em ST . . . . .	29
3.18	Controlador - Implementação em ST . . . . .	29
3.19	Nível de Controlo - Implementação em ST e FDB . . . . .	30
4.1	Arquitectura. . . . .	32
4.2	Processo laboratorial . . . . .	33
4.3	Sensor de nível - tanque 1 . . . . .	34
4.4	Sensor de caudal - tanque 2 . . . . .	34
4.5	Electroválvula. . . . .	35
4.6	Válvula manual. . . . .	35
4.7	Processo laboratorial - Diagrama funcional . . . . .	35
4.8	Autómato controlador. . . . .	36
4.9	Autómato supervisor. . . . .	36
4.10	Estado 1: $y_1 > h_p$ e $y_2 > h_p$ . . . . .	37
4.11	Estado 2: $y_1 > h_p$ e $y_2 < h_p$ . . . . .	37
4.12	Estado 3: $y_1 < h_p$ e $y_2 > h_p$ . . . . .	37
4.13	Estado 4: $y_1 < h_p$ e $y_2 < h_p$ . . . . .	37
4.14	Falha 1: Abertura de $EV_1$ . . . . .	39
4.15	Falha 2: Abertura de $EV_2$ . . . . .	39
4.16	Falha 4: Abertura de $EV_3$ . . . . .	39
4.17	Falha 3: Abertura de $EV_1$ e $EV_2$ . . . . .	39

4.18	Dinâmica do sistema - aproximação considerada. . . . .	40
4.19	Diagrama de controlo em cascata. . . . .	40
4.20	Módulo de processamento[0]; Módulo analógico[1]; Módulo digital[2] . . . . .	41
4.21	Definição dos parâmetros de rede . . . . .	41
4.22	Diagrama de fluxo. . . . .	43
4.23	Aquisição de dados. . . . .	45
4.24	Comunicação. . . . .	45
4.25	Conversão de variáveis de detecção. . . . .	46
4.26	Conversão de variáveis de correcção. . . . .	47
4.27	Estrutura de controlo - FBD. . . . .	47
4.28	Actuação - FBD. . . . .	48
4.29	Ensaio nominal . . . . .	49
4.30	Ensaio - Falha 1 . . . . .	50
4.31	Ensaio - Falha 2 . . . . .	50
4.32	Ensaio - Falha 3 . . . . .	51
5.1	Circuito electrónico - sensor de nível . . . . .	59
5.2	Sensor de nível . . . . .	59
5.3	Circuito electrónico - sensor de caudal . . . . .	61
5.4	Sensor de caudal . . . . .	61
5.5	Circuito electrónico - electroválvulas . . . . .	63
5.6	Electroválvula . . . . .	63



# Lista de Tabelas

4.1	Variáveis de entrada e saída do módulo analógico. . . . .	42
4.2	Tabela de controlo. . . . .	42
4.3	Estrutura PARAM.PI . . . . .	43
4.4	Parâmetros de controlo. . . . .	43
4.5	Tabela de comunicação controlador → supervisor. . . . .	44
4.6	Tabela de comunicação supervisor → controlador. . . . .	44
4.7	Tabela auxiliar de detecção e correcção. . . . .	44





# Acrónimos

- **FBD:** (Functional Block Diagram) Diagrama Funcional de Blocos
- **FDD:** (Fault Detect and Diagnosis) sistema de Detecção e Diagnóstico de Falhas
- **FTC:** (Fault Tolerant Control) Controlo Tolerante a Falhas
- **HMI:** (Human-Machine Interface) Interface Homem-Máquina
- **IL:** (Instruction List) Lista de Instruções
- **LD:** (Ladder Diagram) Diagrama de Escada
- **MRAC:** (Model-Reference Adaptative Control) Controlo Adaptativo por Modelo de Referência
- **P:** controlador Proporcional
- **PI:** controlador Proporcional Integral
- **PID:** controlador Proporcional Integral Derivativo
- **PLC:** (Programmable Logic Controller) Controlador Lógico Programável
- **SFC:** (Sequential Functional Chart) Grafo Funcional Sequencial
- **ST:** (Structured Text) Texto Estruturado
- **STC:**(Self Tuning Control) Controlo auto-sintonizável



# Lista de Simbolos

- $d_{Actuadores}$ : Desvio dos actuadores;
- $d_{Sensores}$ : Desvio dos sensores;
- $f_{Actuadores}$ : Falhas nos actuadores;
- $f_{Processo}$ : Falhas no processo;
- $f_{Sensores}$ : Falhas nos sensores;
- $h_1$ : Nível do tanque 1;
- $h_2$ : Nível do tanque 2;
- $i_{Processo}$ : Identificação de falha no processo;
- $I$ : Componente Integral;
- $I_n$ : Componente integral no instante k;
- $I_0$ : Componente integral no instante k-1;
- $K_d$ : Ganho derivativo;
- $K_i$ : Ganho integral;
- $K_p$ : Ganho proporcional;
- $K_{pn}$ : Ganho proporcional no instante k;
- $K_{po}$ : Ganho proporcional no instante k-1;
- $q_e$ : Caudal de entrada;
- $q_s$ : Caudal de saída;
- $q_{12}$ : Caudal entre reservatórios.
- $T_i$ : Tempo de integração;
- $T_d$ : Tempo de amostragem;
- $u$ : Sinal de controlo não corrigido;

- $u_a$ : Acção de controlo;
- $u_c$ : Sinal de controlo corrigido;
- $y$ : Saídas dos sensores corrigidas;
- $y_p$ : Saídas do processo;
- $y_s$ : Saídas dos sensores;
- $y_1$ : Nível do tanque 1;
- $y_2$ : Nível do tanque 2;

# Capítulo 1

## Introdução

Neste capítulo introdutório é feito um enquadramento do trabalho que resultou nesta tese, referindo-se a importância do desenvolvimento de trabalhos de investigação nas áreas de controlo tolerante a falhas e automação. Para além da relevância das duas áreas de estudo, pretende-se demonstrar a sua complementaridade e os benefícios que advêm do estudo integrado de ambas.

Na secção 1.2 deste capítulo são mencionadas as motivações para a realização deste projecto, sendo estas essencialmente a importância que a investigação nestas áreas tem no desenvolvimento de instalações industriais mais automatizadas e consequentemente mais seguras e eficazes.

Na secção 1.3 são identificados os objectivos e contribuições que este trabalho em particular pode trazer para as áreas de estudo em questão. De um modo geral, este trabalho propõe uma metodologia de controlo tolerante a falhas e uma metodologia para a sua implementação em autómatos programáveis, contribuindo assim para o desenvolvimento conjunto das áreas de controlo e automação.

Na secção 1.4 são referidos os principais aspectos do sistema de controlo tolerante a falhas proposto, mencionando as metodologias utilizadas para o seu dimensionamento.

Este capítulo introdutório é concluído com a secção 1.5 onde é apresentada a estrutura deste documento, sendo feito um resumo dos pontos tratados em cada um dos cinco capítulos que o compõem.

## 1.1 Introdução

As crescentes exigências de segurança e dos requisitos de qualidade para os produtos industrialmente produzidos traduzem-se num aumento da complexidade dos processos de produção, tornando as instalações industriais cada vez mais complexas e automatizadas. Consequentemente, obtêm-se instalações com um maior número de componentes, aumentando assim a probabilidade de ocorrência de falhas. A alteração da dinâmica de funcionamento de uma instalação ou a avaria de sensores ou actuadores pode resultar na alteração significativa das características do produto final produzido ou pôr em perigo a integridade da instalação industrial, incluindo equipamento e trabalhadores. Surge portanto a necessidade de desenvolver métodos capazes de evitar que a falha de um ou vários componentes degrade o desempenho global do sistema.

Genericamente, uma instalação industrial pode ser descrita pelo conjunto de três elementos - sensores, actuadores e processo. O processo é definido pelo conjunto dos componentes da instalação industrial (sistemas de tanques, campos de painéis solares, centrais termoeléctricas, etc.) cujo funcionamento obedece a um conjunto de dinâmicas de complexidade e linearidade variável e que caracterizam o seu comportamento. Os sensores são elementos instalados no processo e responsáveis pela aquisição das suas grandezas mensuráveis, permitindo determinar o seu estado de funcionamento (sensores de temperatura, nível, pressão, etc.). Os actuadores são componentes cuja acção permite modificar o estado do processo podendo ser accionados de forma automática ou manual (motores, válvulas, compressores, etc.).

Todos estes elementos estão sujeitos a falhas de severidade variável existindo diversas formas de as contornar ou minimizar os seus impactos no sistema. Para os casos em que as falhas ocorrem nos sensores é possível a utilização de componentes redundantes, ou seja, com a mesma função. No entanto esta solução pode revelar-se economicamente dispendiosa ou mesmo impraticável. Nestas situações utiliza-se frequentemente a redundância analítica, ou seja, a determinação das grandezas mensuráveis da instalação é realizada de forma indirecta e através de métodos analíticos.

As falhas nos actuadores podem igualmente ser resolvidas através da utilização de componentes redundantes, que, como foi referido anteriormente, pode verificar-se impraticável por motivos económicos e de praticabilidade. No entanto, se a falha não for total e irreversível, ou seja, não afectando o funcionamento do actuador, reflectindo-se apenas na relação entre o sinal e a acção de controlo, é possível acomodar a falha através de métodos analíticos.

Para os casos em que a falha ocorra em componentes do processo, alterando significativamente a sua dinâmica de funcionamento, surge na maioria das vezes a necessidade de alterar também a dinâmica de controlo de forma a acomodar as falhas e manter o desempenho do sistema dentro dos padrões satisfatórios. Noutros casos, um sistema de controlo robusto, cujo dimensionamento permite cobrir toda a gama de funcionamento (nominal e com falhas), pode garantir um desempenho satisfatório.

Em suma, existem diversas abordagens para o dimensionamento de sistemas de controlo tolerante a falhas, sendo necessário analisar a sua severidade e os componentes em que ocorrem, de forma a obter a melhor solução.

## **1.2 Motivações**

A realização deste trabalho é motivada pela importância do desenvolvimento de novas abordagens de controlo tolerante a falhas, de modo a acompanhar a crescente automatização e diversificação das instalações industriais. Novas abordagens conduzem geralmente a métodos analíticos de maior complexidade que necessitam de dispositivos com maior capacidade de processamento. No entanto, as exigências deste tipo de algoritmos de controlo não são muitas vezes suficientes para justificar a sua implementação em computadores, cuja capacidade de processamento é superior à necessária. Deste modo, a utilização de autómatos programáveis para este tipo de sistemas revela-se frequentemente uma solução eficaz do ponto de vista operacional e económico, sendo por isso amplamente utilizados em ambiente industrial. Assim, paralelamente ao desenvolvimento de metodologias de controlo, surge também a necessidade de desenvolver propostas para a sua implementação em autómatos programáveis sendo também este um ponto de motivação para o trabalho realizado.

## **1.3 Objectivos e Contribuições**

Os objectivos e contribuições do trabalho realizado enquadram-se em duas áreas de estudo distintas - controlo tolerante a falhas e automação - sendo o objectivo principal desta tese apresentar uma proposta que integre as duas áreas. Deste modo, o trabalho realizado pretende apresentar uma metodologia de controlo tolerante a falhas válida, capaz de garantir o desempenho do sistema em caso de falha nos sensores, processo ou actuadores, sugerindo paralelamente uma proposta de implementação em autómatos programáveis para as estruturas de controlo apresentadas.

## **1.4 O Sistema Proposto de Controlo Tolerante a Falhas**

O sistema proposto enquadra-se numa metodologia activa de controlo tolerante a falhas que se caracteriza pela reconfiguração ou alteração dos parâmetros de controlo de acordo com a natureza e severidade da falha. A abordagem seguida propõe que a alteração dos parâmetros se efectue segundo um conjunto de regras e parâmetros previamente projectados de forma a cobrir todas as falhas, enquadrando-se assim numa metodologia baseada em métodos de projecção.

Esta abordagem pressupõe a existência de uma estrutura de detecção e diagnóstico capaz de identificar correctamente a falha, sendo através dessa identificação, tomada a decisão de quais os

parâmetros de controlo a utilizar. Deste modo, o sistema de detecção e diagnóstico de falhas é um componente essencial mas cujo funcionamento se encontra fora do âmbito desta tese.

## 1.5 Organização da Tese

Para além deste primeiro capítulo introdutório, onde se pretende enquadrar o trabalho realizado, este documento é constituído por mais quatro capítulos. Os seus temas e pontos de interesse são sumarizados nos parágrafos seguintes.

No capítulo dois é realizado um levantamento do estado da arte nas áreas de estudo em que este trabalho se insere - controlo tolerante a falhas e automação. Na primeira secção deste capítulo são descritas as duas metodologias de controlo tolerante a falhas - activa e passiva - referindo as várias abordagens existentes dentro de cada metodologia, analisando as suas características, vantagens, desvantagens e aplicabilidade. Na segunda secção apresenta-se as cinco linguagens para a programação de autómatos, descrevendo as suas características e sintaxe. O capítulo termina com um enquadramento e análise comparativa entre a proposta apresentada e as diversas metodologias existentes.

No terceiro capítulo é apresentado em pormenor o sistema de controlo proposto, sendo descrito detalhadamente as suas metodologias, características e estruturas internas que o compõem. No final do capítulo são apresentadas as propostas para a sua implementação em autómatos, utilizando essencialmente as linguagens Texto Estruturado e Diagrama Funcional de Blocos.

No capítulo quatro é descrito o caso de estudo onde foi testado o sistema de controlo proposto, começando por referir detalhadamente os componentes que compõem o processo laboratorial e as suas dinâmicas de funcionamento. Em seguida são descritos os pontos principais da implementação nos autómatos *Modicon M320*, referindo as etapas necessárias à configuração do hardware e definição de variáveis. No final deste capítulo são apresentados e discutidos os resultados obtidos nos ensaios realizados a fim de aferir o desempenho do sistema.

No quinto e último capítulo deste documento são apresentadas as conclusões retiradas ao longo da realização deste trabalho, analisando o desempenho, viabilidade, vantagens e desvantagens da proposta apresentada. Adicionalmente são sugeridos vários pontos para trabalho futuro.



## Capítulo 2

# Estado de Arte

Neste capítulo são apresentadas algumas das metodologias mais estudadas no controlo tolerante a falhas, referindo as suas características e aplicabilidade. Genericamente, o controlo tolerante a falhas pode ser classificado em duas abordagens - activa e passiva. A abordagem passiva é baseada em técnicas de controlo robusto cujo objectivo é desenvolver controladores capazes de manter o desempenho do sistema na presença de falhas, sem recorrer no entanto à reconfiguração ou alteração dos parâmetros de controlo. Em oposição, as abordagens activas propõem que a acomodação das falhas se efectue através da reconfiguração ou alteração dos parâmetros de controlo de acordo com a natureza e severidade da falha.

Na segunda secção deste capítulo faz-se uma introdução ao tema Autómatos Programáveis, referindo os principais pontos da sua arquitectura, características de hardware, software e funcionamento. Em seguida, procede-se à apresentação da norma 61311 do IEC, focando em particular a parte III, onde são descritas as cinco linguagens de programação para autómatos - Texto Estruturado, Diagrama Funcional de Blocos, Diagrama de Escada, Lista de Instruções e Grafo Funcional Sequencial.

## 2.1 Controlo Tolerante a Falhas

A realização de um sistema de controlo tolerante a falhas segue diversas abordagens. Enquanto umas assentam em técnicas de controlo robusto - abordagem passiva - existem outras que propõem a modificação das estruturas de controlo de forma a acomodar a falha - abordagem activa. Dentro da abordagem activa existem ainda várias propostas para a reconfiguração do sistema de controlo, que, genericamente, se agrupam em métodos de projecção ou de reconfiguração online. Na figura 2.1 é apresentada uma visão global sobre as diversas abordagens, seguindo-se na primeira secção deste capítulo uma descrição das suas características, vantagens e desvantagens.

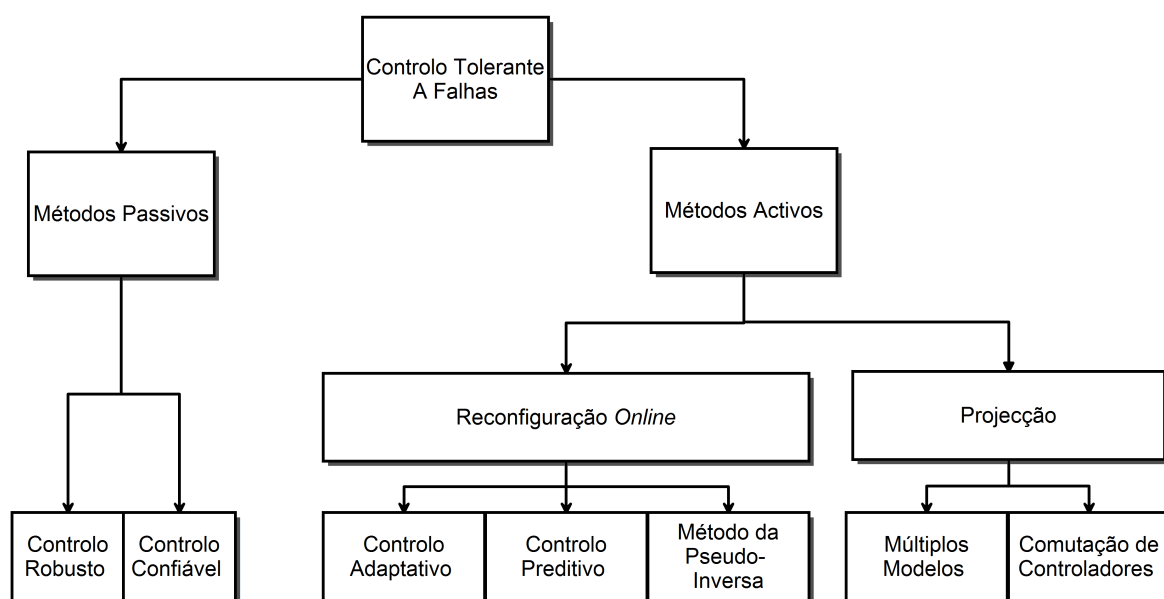


Figura 2.1: Controlo Tolerante a Falhas - Principais Metodologias

### 2.1.1 Métodos Passivos de Controlo Tolerante a Falhas

Como já referido anteriormente, a abordagem passiva baseia-se em técnicas de controlo robusto que permitem criar insensibilidade a determinadas falhas, mantendo o desempenho do sistema dentro de parâmetros satisfatórios. No entanto, devido às suas características não-adaptativas, a aplicabilidade desta abordagem está limitada a condições específicas.

A robustez e insensibilidade a falhas está restrita a um número relativamente pequeno de casos, sendo apenas eficaz em falhas que não afectem de forma significativa o comportamento do sistema. No entanto, de forma a alargar o espectro de falhas a que o sistema de controlo responde de forma satisfatória, recorre-se muitas vezes a um aumento da robustez e insensibilidade do controlador que, na maioria dos casos, traduz-se também na degradação do desempenho do sistema quando este se encontra em funcionamento nominal. Considerando que a maioria das falhas são acontecimentos momentâneos e esporádicos, o aumento da robustez em troca da diminuição do desempenho em funcionamento nominal pode, em determinados casos, não ser uma solução razoável. No entanto,

por não necessitar de um sistema complementar de detecção e diagnóstico de falhas e assentar numa estrutura de controlo única e inalterável, revela-se uma abordagem atractiva do ponto de vista de implementação prática e por isso uma boa solução em determinados casos.

### **Controlo Confiável**

A abordagem de projecto de controladores confiáveis têm como objectivo principal garantir o desempenho do sistema, assegurando a estabilidade e o desempenho pretendido na presença de falhas previstas. O objectivo traduz-se na procura de um controlador que optimize o desempenho com a pior-falha para todas as falhas previstas. Como referência para métodos de projecto de controladores confiáveis, destacam-se os artigos de Ferreira (2002), Hsieh (2002), Liao *et al.* (2002), Niemann and Stoustrup (2002), Suyama (2002) e Cardoso (2006)

### **Controlo Robusto**

O projecto de controladores robustos baseia-se numa outra classe de abordagens passivas em que se pretende obter um controlador que satisfaça as especificações de projecto em condições normais de funcionamento do sistema e que garanta um desempenho satisfatório na presença de falhas. Estas abordagens são usualmente baseadas na teoria por realimentação quantitativa ou na teoria do controlo robusto em  $H^\infty$ . Como referência para métodos de projecto de controladores robustos, destacam-se os artigos de Fliess *et al.* (2004; 2005) e Cardoso (2006)

## **2.1.2 Métodos Activos de Controlo Tolerante a Falhas**

As abordagens activas de controlo tolerante a falhas têm, como já referido anteriormente, por princípio de funcionamento a reconfiguração ou a modificação dos parâmetros de controlo em caso de ocorrência de falha. Ao contrário das abordagens passivas, estes métodos necessitam geralmente de um sistema complementar de detecção e diagnóstico que realize a detecção, isolamento e identificação das falhas. A arquitectura de um sistema activo de controlo tolerante a falhas com um sistema complementar de detecção e diagnóstico de falhas pode ser genericamente representada como se ilustra na figura 2.2.

A metodologia adoptada para a reconfiguração do sistema de controlo pode ser enquadrada em duas linhas distintas - métodos de projecção ou métodos de reconfiguração em linha. Os métodos de projecção têm por base a selecção de um controlador, ou conjunto de parâmetros, previamente dimensionados de forma a cobrir toda a gama de funcionamento (com e sem falhas), estando por isso a sua utilização restrita a casos em que as falhas pertencem a um grupo conhecido e finito. Em oposição, os métodos baseados na reconfiguração em linha assentam na determinação e modificação da estrutura de controlo com o sistema em funcionamento, conseguindo desta forma acomodar um número maior

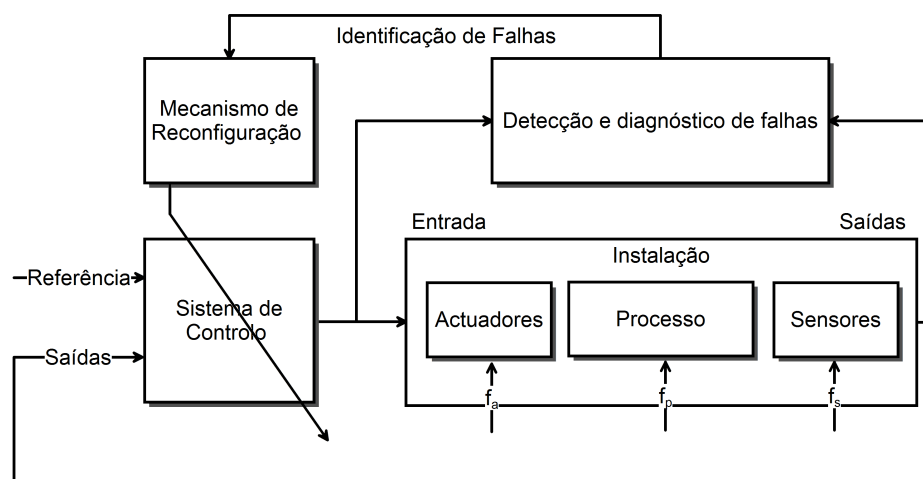


Figura 2.2: Métodos Activos de Controlo Tolerante a Falhas - Arquitectura geral

de falhas. Comparativamente aos métodos de projecção, a reconfiguração em linha envolve algoritmos de controlo significativamente mais complexos necessitando por isso de dispositivos com maior capacidade de processamento.

Um aspecto relevante na concepção deste tipo de sistemas é a interacção entre o sistema de controlo e o sistema de detecção e diagnóstico de falhas. A maioria das abordagens foca-se no projecto do sistema de controlo, considerando perfeito o funcionamento do sistema de detecção e diagnóstico de falhas. No entanto esta consideração pode deixar de lado aspectos importantes, como por exemplo, a identificação incorrecta, detecção com atraso ou não detecção de falhas.

Por conseguirem abranger um espectro maior de casos, os métodos activos de controlo tolerante a falhas são actualmente um assunto de estudo actual. Nos parágrafos seguintes são descritos os métodos mais comuns.

## Múltiplos Modelos

O método dos múltiplos modelos é uma abordagem que assenta na descrição da dinâmica do processo através de um conjunto finito de modelos lineares, obtidos de forma a abranger toda a gama de funcionamento (nominal e com falhas). Adicionalmente, para cada um dos modelos é projectado um controlador adequado. O dimensionamento dos modelos e controladores é realizado *offline* sendo por isso esta abordagem enquadrada nos métodos de projecção.

A acção de controlo resulta de uma combinação ponderada das várias acções de controlo geradas pelos diversos controladores, sendo o peso de cada controlador determinado de acordo com a aproximação entre o respectivo modelo e o processo. Na figura 2.3 é representada a arquitectura genérica de um sistema de controlo tolerante a falhas baseado no método dos múltiplos modelos. Como referências para o método de múltiplos modelos destacam-se os trabalhos de Zhang and Jiang (2001), Demetriou (2001), Theilliol *et al.* (2003), Yen and Ho (2003) e Cardoso (2006).

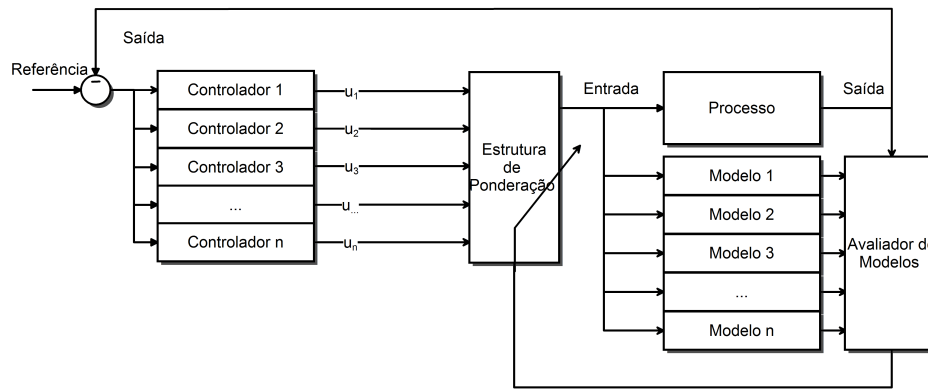


Figura 2.3: Método de Múltiplos Modelos - Arquitectura

Para o funcionamento nominal ou com as falhas previstas, é previsível que acção de controlo seja na sua maioria determinada por um controlador, sendo praticamente nula a influência dos restantes. No entanto, para falhas não previstas poderá, verificar-se a influência de diversos controladores, cuja acção conjunta pode levar à instabilidade do sistema, traduzindo-se por isso numa desvantagem deste método. Deste modo, embora constitua uma boa solução para processos com dinâmicas não lineares, o método dos múltiplos modelos é apenas indicado para casos em que as falhas pertencem a um conjunto restrito e finito.

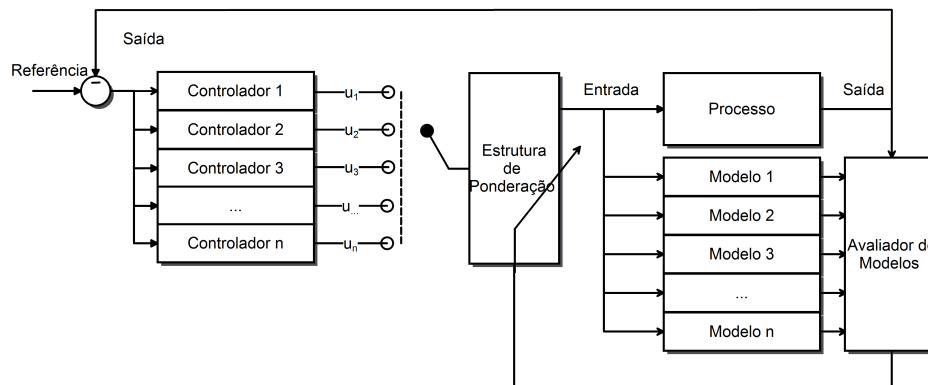


Figura 2.4: Método de Comutação de Controladores - Arquitectura

### Comutação de Controladores

O método de comutação de controladores, à semelhança do método de múltiplos modelos, considera um conjunto de modelos e controladores que abrangem o comportamento do processo para as situações nominais e de falha. No entanto, ao contrário do método anterior, a acção de controlo é determinada apenas pela acção de um único controlador e não pela acção conjunta e ponderada de vários. A determinação do controlador activo é igualmente obtida através da avaliação do modelo que melhor reflecte o comportamento instantâneo do processo. Genericamente, a arquitectura de um sistema de controlo tolerante a falhas baseado no método de comutação de controladores é a apresentada na figura 2.4. Como referências para o método de comutação de controladores destacam-se os trabalhos de Chang *et al.* (2001), Maki *et al.* (2001), Médar *et al.* (2002a, 2000b) e Cardoso (2006).

## Controlo Preditivo

O método de controlo preditivo assenta em técnicas de controlo óptimo cuja formulação do problema de optimização incorpora a referência, saída predita e restrições sobre as entradas, saídas e estados do sistema. Deste modo, as alterações (devido a falha) do funcionamento de sensores, actuadores ou processo, podem ser acomodadas através da reformulação das restrições do problema de optimização. Este método necessita portanto de um sistema de detecção e diagnóstico complementar que, através da identificação da natureza e severidade da falha, deverá fornecer a actualização das restrições e modelo preditivo. Na figura 2.5 é apresentada arquitectura genérica de um sistema de controlo tolerante a falhas baseada no método de controlo preditivo.

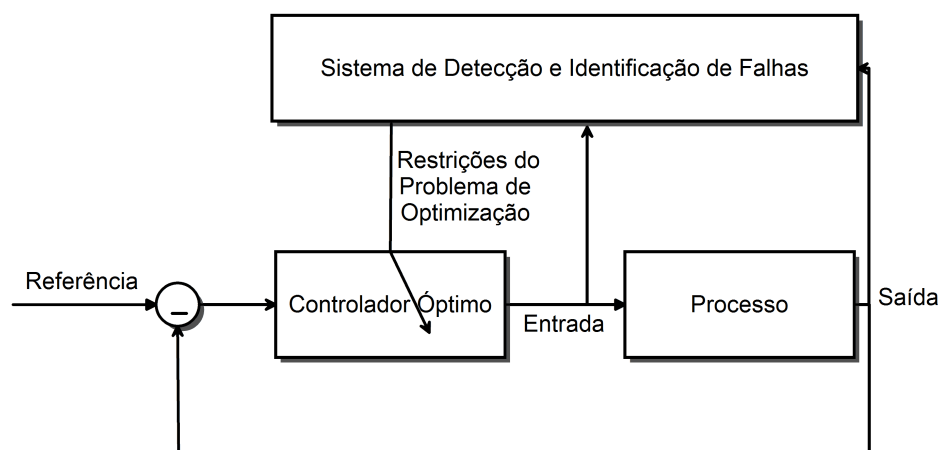


Figura 2.5: Método de Controlo Preditivo - Arquitectura

Este método tem o seu uso generalizado em ambiente industrial por assentar em princípios de controlo relativamente simples, facilitando assim a sua implementação. No entanto, a execução do processo de optimização em cada passo temporal torna a utilização deste método adequada apenas a processos com dinâmicas temporais lentas, como por exemplo processos químicos ou de energia solar. Como referências para o método de controlo preditivo destacam-se os trabalhos de Gil *et al.* (2003), Aström *et al.* (2001) e Cardoso (2006).

## Controlo Adaptativo

De modo a tornar um sistema tolerante a falhas, os métodos de controlo adaptativo propõem o ajuste automático dos parâmetros de controlo, existindo duas abordagens distintas para referida adaptação - directa e indirecta. Na abordagem indirecta o reajuste dos parâmetros de controlo é realizado através dos parâmetros do processo, sendo por isso necessária a sua determinação. Em oposição, na abordagem directa o controlador é ajustado através da comparação entre a saída do processo e a trajectória requerida, sem recorrer à estimação de parâmetros do processo. Os métodos mais comuns de controlo adaptativo *Model-Reference Adaptive Control* (MRAC) e *Self Tuning Control* (STF). As arquitecturas genéricas de um sistema de controlo tolerante a falhas através de controlo adaptativo é ilustrada na figuras 2.6 e 2.7. Como referência para o método de controlo

adaptativo destacam-se os trabalhos de Dionísio *et al.* (2003), Jiang *et al.* (2003) e Cardoso (2006).

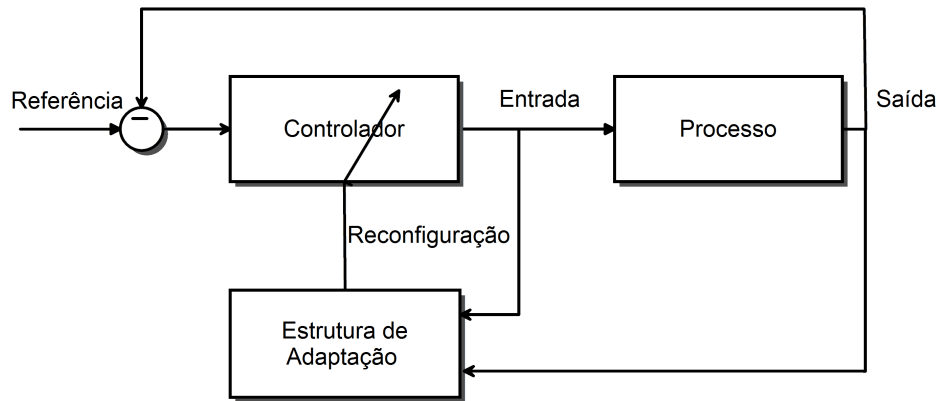


Figura 2.6: Método de Controlo Adaptativo - abordagem directa.

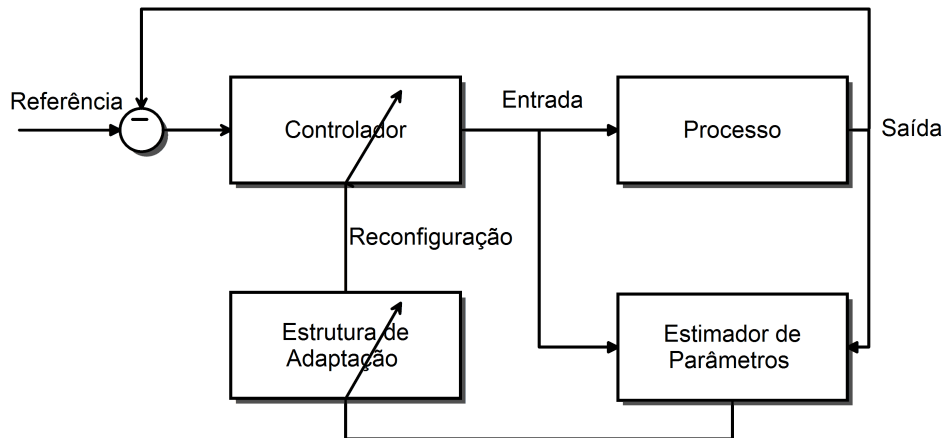


Figura 2.7: Método de Controlo Adaptativo - abordagem indirecta.

## 2.2 Autómatos Programáveis

Os autómatos programáveis, ou PLCs (*Programmable Logic Controller*), assumem actualmente um papel preponderante nas instalações industriais, tendo o seu uso sido generalizado a diversas áreas de automação. Estes dispositivos de processamento apresentam características de hardware robustas o que permite a sua utilização em ambientes severos, onde estão expostos a, por exemplo, altas e baixas temperaturas, poeiras, vibrações etc. Adicionalmente, os PLCs apresentam grande insensibilidade ao ruído eléctrico, um fenómeno frequente em instalações industriais que deriva do elevado número de máquinas eléctricas em funcionamento e tem efeitos indesejados no processamento dos sinais inerentes ao sistema de controlo.

Para além das características de hardware robustas, os PLCs dispõem de sistemas operativos simples e exclusivamente dedicados à execução do programa de controlo, estando por isso menos susceptíveis a interrupções de processamento inesperadas que podem interferir com as características temporais do sistema, como por exemplo o tempo de amostragem. A arquitectura genérica de um autómato é ilustrada na figura 2.8.

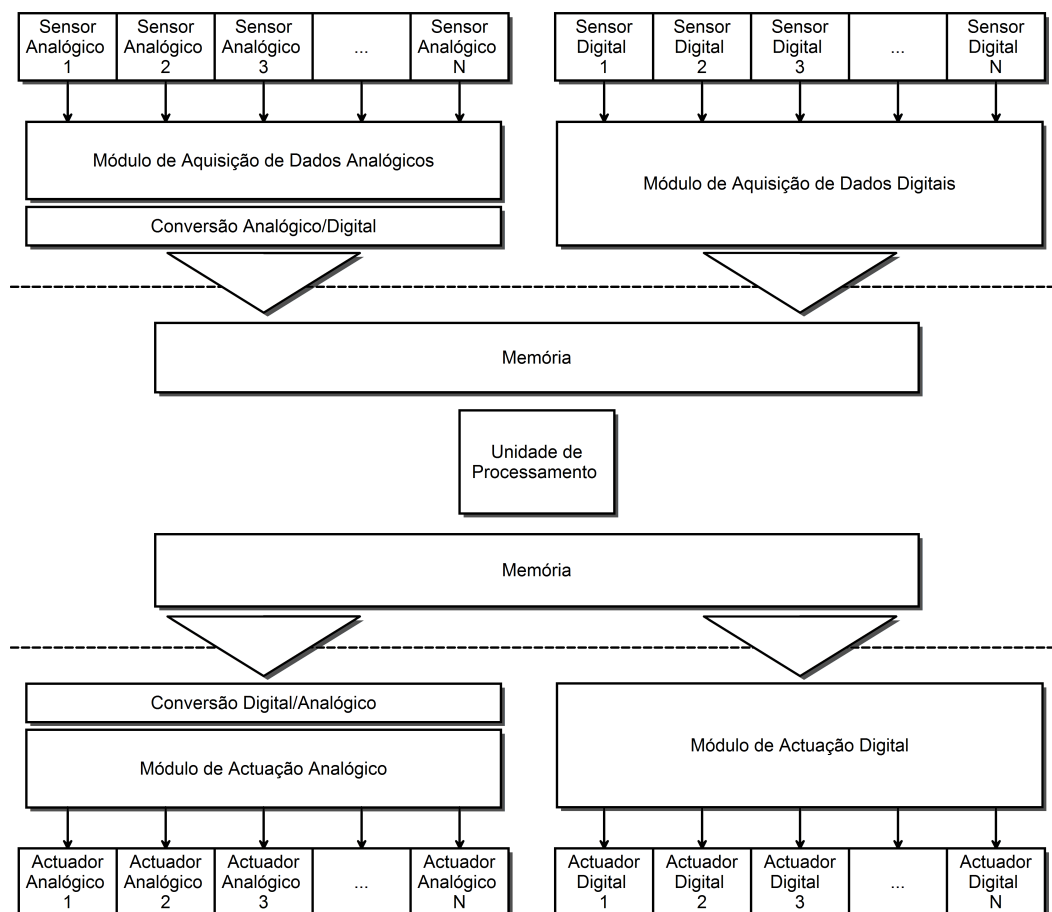


Figura 2.8: Autómatos Programáveis - Arquitectura

A execução do programa é realizada em três partes - aquisição de dados, processamento e actualização das saídas - que se executam repetidamente e por esta ordem. Na primeira etapa é efectuada a aquisição de dados, que tipicamente corresponde à leitura de sensores analógicos ou ao estado de contactores, seguindo-se a conversão de sinais analógicos em digitais e o seu armazenamento na memória interna do dispositivo. Na etapa de processamento é realizado um conjunto de operações aritméticas características da lei de controlo implementada, cujo objectivo final é a determinação do valor das saídas. Finalmente, as saídas são actualizadas com os valores obtidos na etapa de processamento, estando estas geralmente conectadas a actuadores (motores eléctricos, actuadores pneumáticos, etc.).

A necessidade de comunicar com outros autómatos ou dispositivos de interface humano-máquina (*Human Machine Interface* - HMI) levou à necessidade de inclusão e desenvolvimento de portas e protocolos de comunicação, destacando-se entre as várias tecnologias existentes a *Ethernet*, *ModBus* e *CANOpen*.

O aumento do número de instalações industriais automatizadas conduziu a um aumento da utilização de autómatos programáveis, tornando a sua produção economicamente atractiva para muitos fabricantes que iniciaram o desenvolvimento de dispositivos com características de hardware e software próprias. No entanto, a não uniformidade traduz-se geralmente em problemas de integração, tendo surgido por isso a necessidade de definir algumas regras sobre a utilização destes dispositivos. Com



este objectivo, o IEC (*International Electrotechnical Commission*) formulou em 1993 a norma 61311, constituída por oito partes, sendo a terceira parte dedicada à definição de algumas directrizes para as linguagens de programação em autómatos - Texto Estruturado, Diagrama Funcional de Blocos, Diagrama de Escada, Lista de Instruções e Grafo Funcional Sequencial. Nos parágrafos seguintes segue-se uma breve descrição e exemplo de cada uma das cinco linguagens.

### 2.2.1 Diagrama em Escada

O diagrama em escada é uma linguagem que deriva dos diagramas de circuitos eléctricos, sendo convencionalmente utilizada para representar circuitos de lógica combinatória. Por não dispor de operadores aritméticos, o LD (*Ladder Diagram*) tem a sua utilização restrita a um número reduzido de casos, sendo no entanto uma solução viável para aplicações que assentem essencialmente em lógica booleana. Na figura 2.9 é ilustrado um exemplo de aplicação do LD.

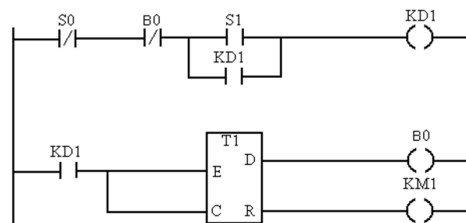


Figura 2.9: Diagrama de Escada - Exemplo

### 2.2.2 Lista de Instruções

A lista de instruções é uma linguagem de baixo nível com estrutura semelhante à linguagem de máquina *assembly*. O IL (*Instruction List*) é caracterizado por uma sintaxe simples, tendo por isso a sua capacidade de utilização limitada a problemas de reduzida complexidade. Dispõem de operadores que permitem transitar entre partes não consecutivas do programa, tornando a sua execução descontínua. A semelhança do IL com o *assembly* leva a que alguns construtores de PLC façam do IL a linguagem de programação base para a qual todas as outras linguagens são convertidas. Na figura 2.10 é ilustrado um exemplo de aplicação do IL.

```
LD      SM 0.1
MOVD   # 4000, VD200
LD      SM 0.1
MOVW   # 41, VW10
LD      SM 0.1
DIV    VW10, VD200
MEND
```

Figura 2.10: Lista de Instruções - Exemplo

### 2.2.3 Grafo Funcional Sequencial

O grafo funcional sequencial é uma linguagem gráfica que tem a sua origem no *Grafcet*, apresentando uma simbologia e metodologia de descrição semelhante. O SFC (*Sequential Function Chart*) representa um sistema em termos dos seus diferentes estados, sendo o fluxo entre estes controlado através de transições condicionais que podem ser definidas através de, por exemplo, temporizadores, lógica combinatória, variáveis do sistema, etc. Na figura 2.11 é ilustrado um exemplo de aplicação do SFC.

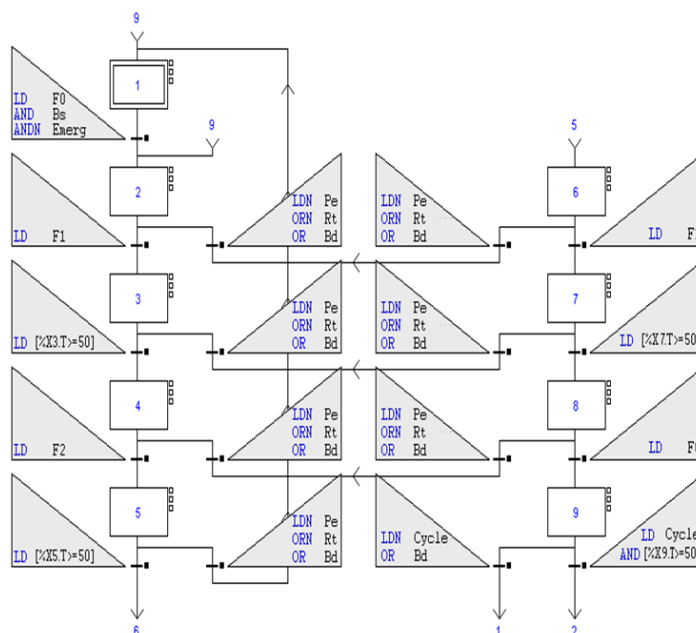


Figura 2.11: Grafo Funcional Sequencial - Exemplo

#### 2.2.4 Texto Estruturado

O Texto estruturado é uma linguagem de alto nível com sintaxe semelhante ao *PASCAL* tendo sido inicialmente desenvolvida para a resolução de problemas em tempo real. A norma define-a como uma linguagem constituída por declarações que permitem atribuir valores a variáveis de diversos tipos dispondo de vários operadores para tal - aritméticos, booleanos, de comparação, execução condicional e ciclos iterativos. O ST (*Structured Text*) permite a evocação de funções auxiliares durante a execução do programa principal, o que possibilita a organização do programa em diversos módulos, conferindo-lhe assim maior legibilidade. Nas figuras 2.12 a 2.15 são ilustrados exemplos de aplicação do ST.

```
IF A>0 THEN
    B:=1;
ELSIF A<0 THEN
    B:=-1;
ELSE
    B:=0;
END IF;
```

```
FOR I:=0 TO 10 BY +1 DO
    B:=B+1;
END_FOR;
WHILE C<10 DO
    C:=C+1;
END WHILE;
```

Figura 2.12: Texto Estruturado - Exemplo 1.

Figura 2.13: Texto Estruturado - Exemplo 2.

```

REPEAT
    D:=D+1;
UNTIL D>10
END_REPEAT;

```

Figura 2.14: Texto Estruturado - Exemplo 3.

```

CASE E OF
    0: F:=-1;
    1: F:=1;
ELSE
    F:=0;
END_CASE;

```

Figura 2.15: Texto Estruturado - Exemplo 4.

### 2.2.5 Diagrama Funcional de Blocos

O diagrama funcional de blocos é a primeira linguagem gráfica descrita na norma que define um conjunto de diretrizes para os símbolos do FBD (*Function Block Diagram*). À semelhança de um circuito electrónico, esta linguagem assenta na descrição de um sistema através do fluxo de sinais entre os vários componentes, representados por blocos gráficos, cujo comportamento pode ser descrito através de outras linguagens. É idealmente utilizado para descrever comportamentos contínuos, necessitando de ferramentas auxiliares para comportamentos sequenciais. Na figura 2.16 é ilustrado um exemplo de aplicação do FBD.

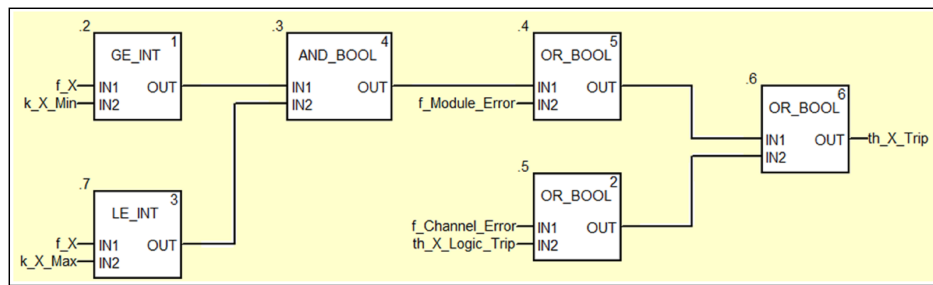


Figura 2.16: Diagrama Funcional de Blocos - Exemplo



## Capítulo 3

# Sistema de Controlo Proposto

Na primeira secção deste capítulo é apresentada a arquitectura global do sistema que se considera subdividida em três níveis - nível de processo, de controlo e de supervisão. O nível de processo é constituído por todos os elementos físicos que compõem a instalação, ou seja, processo, sensores e actuadores. O nível de controlo contém as estruturas analíticas necessárias à regulação das saídas do processo (em funcionamento nominal ou com falhas) - Decisor, Sensores Virtuais, Actuadores Virtuais e Controlador. O nível de supervisão engloba as estruturas destinadas à detecção e diagnóstico de falhas, validação de referências e o próprio supervisor humano.

Na segunda secção referem-se as metodologias consideradas para a realização e dimensionamento de cada um dos elementos que constituem o nível de controlo, referindo detalhadamente o seu funcionamento e os aspectos operacionais da sua implementação em autómatos programáveis.

### 3.1 Arquitectura

O sistema de controlo tolerante a falhas proposto é estruturado em três níveis - nível de processo, de controlo e de supervisão. Na figura 3.1 é ilustrada a arquitectura geral do sistema, identificando os elementos que compõem cada nível e o fluxo de informação entre cada componente. Nas secções seguintes apresenta-se uma descrição detalhada do funcionamento das estruturas consideradas.

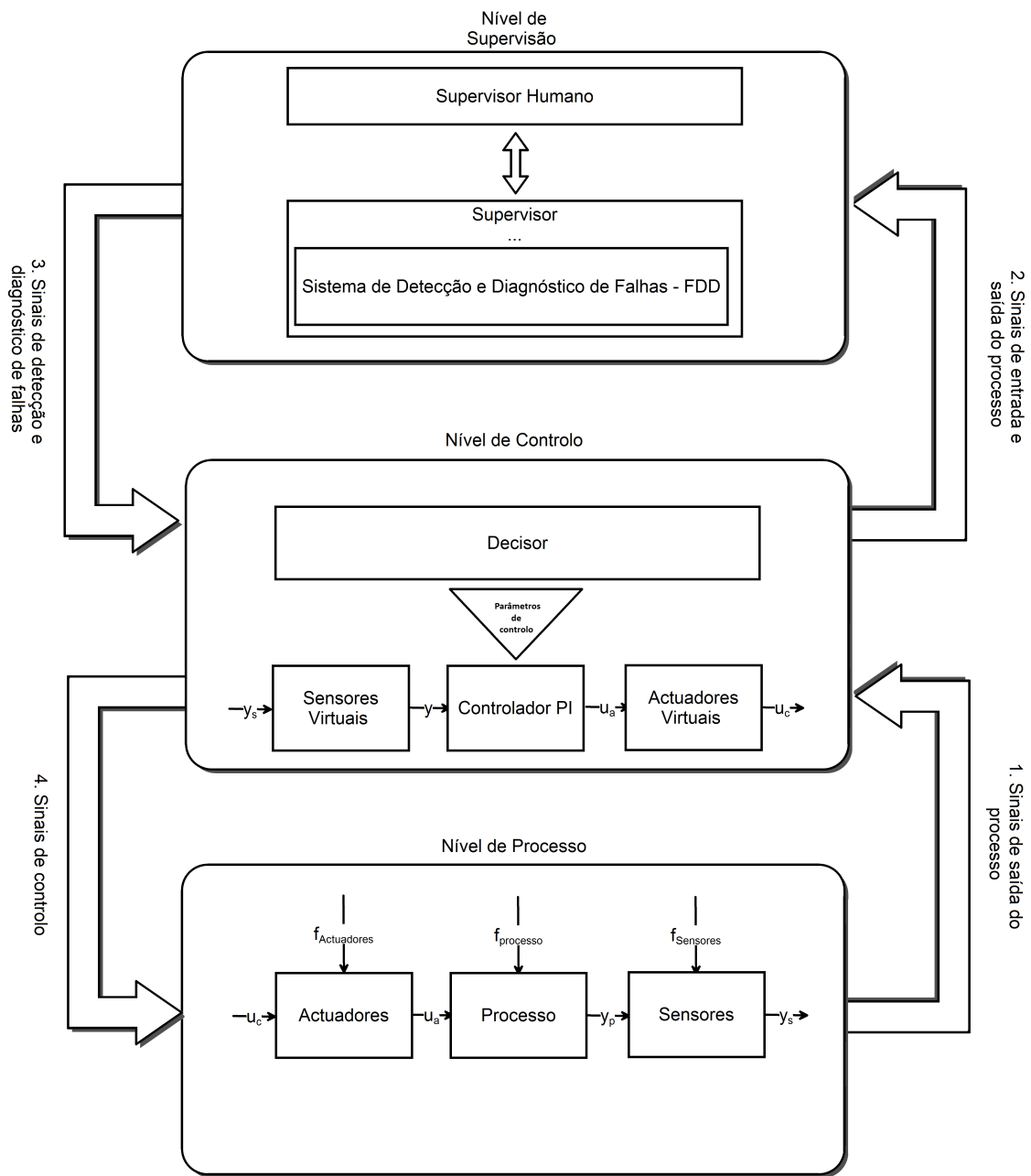


Figura 3.1: Arquitectura de controlo proposta.

### 3.1.1 Nível de Processo

O nível de processo, localizado na base da arquitectura apresentada na figura 3.1, é constituído por três elementos, representados na figura 3.2 - sensores, processo e actuadores - estando cada um destes elementos sujeitos a diversas falhas e perturbações de severidade variável e que alteram a dinâmica global do sistema.

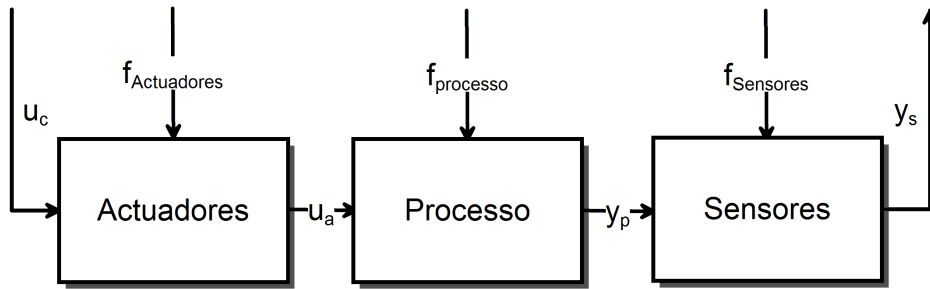


Figura 3.2: Nível de Processo.

- **Processo:** Genericamente, o processo é caracterizado por um conjunto de dinâmicas que relacionam as entradas  $U_a$  com as saídas  $Y_p$ , e que sofrem alterações em caso de ocorrências de falhas no processo  $f_{Processo}$ . As falhas no processo têm geralmente origem no funcionamento defeituoso dos seus componentes, por exemplo, fugas em tanques, entupimentos de tubagens, etc.
- **Actuadores:** Os elementos actuadores aplicam à entrada do processo a acção actuadora  $U_a$  com o objectivo de regular as saídas do processo  $Y_p$ , sendo  $U_a$  regulado pelo sinal de controlo  $U_c$ , proveniente do nível intermédio representado na figura 3.1, denominado nível de controlo. Também os actuadores são caracterizados por um conjunto de dinâmicas que relacionam as suas entradas  $U_c$  com as suas saídas  $U_a$ , podendo essas dinâmicas ser alteradas por falhas  $f_{Actuadores}$ .
- **Sensores:** Os elementos sensores têm o objectivo de obter os valores das saídas mensuráveis do processo  $Y_p$  e fornecer estes dados ao nível de controlo. A aquisição dos valores  $Y_p$  é realizada através de transdutores que relacionam as grandezas físicas  $Y_p$  com os valores  $Y_s$ , sendo também essa relação afectada por falhas  $f_{Sensores}$ .

Em suma, o nível de processo está sujeito a um conjunto de falhas, constituído por falhas nos sensores ( $f_{Sensores}$ ), nos actuadores ( $f_{Actuadores}$ ) ou no processo ( $f_{Processo}$ ), que corrompem os sinais de aquisição ( $Y_p$ ), de actuação ( $U_a$ ) ou a relação entre as entradas ( $U_a$ ) e saídas ( $Y_p$ ), respectivamente. Deste modo o objectivo do sistema de controlo é apresentar uma solução válida que mantenha o desempenho do sistema para cada uma das situações de falha referidas.

### 3.1.2 Nível de Controlo

O nível intermédio da arquitectura ilustrada na figura 3.3, corresponde ao nível de controlo e inclui os módulos de decisão, sensores virtuais, actuadores virtuais, e estrutura de controlo.

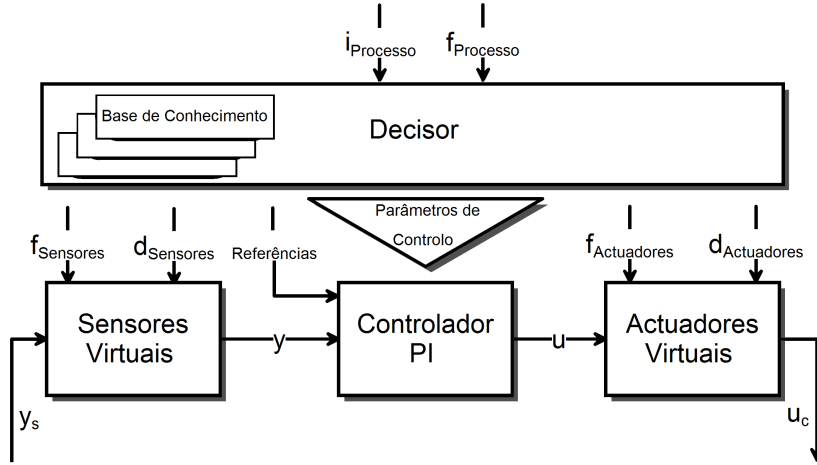


Figura 3.3: Nível de Controlo

Assumindo que a dinâmica do processo, em funcionamento nominal ou com falhas, é representada por um conjunto de modelos lineares com incerteza, a tolerância a falhas no processo é obtida através do dimensionamento de um conjunto de controladores projectados com o objectivo de gerar a acção de controlo,  $u$ , capaz de assegurar de uma forma robusta os requisitos de estabilidade e desempenho, para cada um dos modelos considerados. Os vários controladores projectados consideram-se da mesma estrutura, sendo a sua dinâmica de controlo alterada através da comutação dos seus parâmetros internos. Os parâmetros projectados devem apresentar características robustas, possibilitando algum nível de incerteza nos modelos e no sistema de detecção identificação de falhas.

Com base na detecção e identificação de falhas ( $f_{Processo}$ ), fornecidas pelo nível de supervisão, a estrutura de decisão determina, com base num conjunto de regras pré-definidas, quais os parâmetros de controlo adequados à acomodação da falha. Adicionalmente, o modulo decisor considera a existência de vários modelos lineares para o funcionamento nominal do processo ( $i_{Processo}$ ), seguindo uma metodologia de comutação idêntica à adoptada em caso de falha. A alteração de parâmetros deve ser efectuada tendo em conta vários aspectos da comutação entre controladores, que serão tratados na segunda secção deste capítulo.

Em caso de falha nos sensores, os valores  $y_s$  são corrompidos, não correspondendo aos valores reais da saída do processo  $y_p$ . De forma a acomodar as falhas nos elementos de aquisição, o nível de controlo considera as estruturas sensores virtuais que, com base em redundância analítica, corrigem os valores obtidos  $y_s$ . A detecção de falhas e os valores dos desvios de  $y_s$  em relação a  $y_p$ , são determinados pelo nível de supervisão, com base nos vários modelos que caracterizam a dinâmica do processo. De forma análoga, a tolerância a falhas nos elementos actuadores é obtida através das estruturas actuadores virtuais que, com base na informação do sistema de FDD, corrigem a acção de controlo  $u$ , de forma a que esta coincida com a acção de controlo  $u_a$  aplicada ao processo.



### 3.1.3 Nível de Supervisão

O nível de supervisão, localizado no topo da arquitectura considerada, é ilustrado na figura 3.4. De forma a tornar o sistema global tolerante a falhas, este nível deve cobrir os seguintes aspectos:

- **Interface humano-máquina:** Para o funcionamento global do sistema de controlo é necessário que sejam fornecidas as trajectórias requeridas para as saídas do processo, sendo esta informação definida pelo supervisor humano. Deste modo, para que as decisões do supervisor humano sejam correctas e adequadas ao estado do processo, é necessário que este tenha conhecimento de todas as variáveis do sistema, incluindo acções de controlo, variáveis de estado, sinais de referência e saídas do processo. Na arquitectura proposta considera-se que o nível de supervisão realiza a interface humano-máquina, fornecendo todos os dados relevantes sobre o sistema e recebendo as referências para as saídas do processo.
- **Deteccção e identificação de falhas:** No nível de supervisão considera-se a existência de um sistema de detecccção e identificação de falhas capaz de detectar e isolar alterações no funcionamento do sistema, sendo este um ponto preponderante na tolerância a falhas. Para os casos de falha no processo, para além da detecccção, o nível de supervisão deverá fornecer a identificação da falha, dentro um conjunto previsto. Para os casos de falha nos elementos de aquisição ou actuação, considera-se que o supervisor fornece a correcção dos valores  $y_s$  ou  $u_a$ , respectivamente.

Adicionalmente, em situações em que o sistema de controlo não apresente capacidade de tolerância a uma determinada falha, o supervisor deverá conduzir o processo para uma zona de funcionamento segura, através da definição referências, acções de controlo ou interrompendo o seu funcionamento.

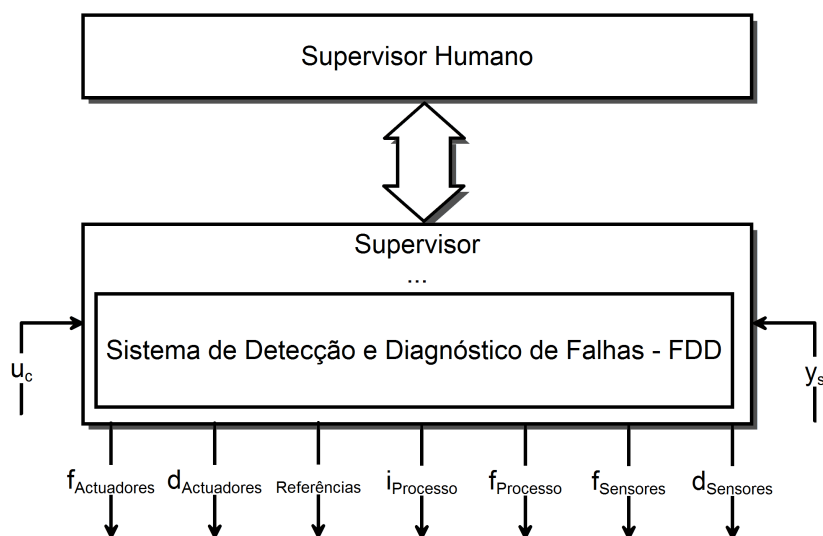


Figura 3.4: Nível de Supervisão.

## 3.2 Metodologia

Nesta secção é feita uma descrição das metodologias adoptadas para a realização do sistema de.

### 3.2.1 Decisor

Como foi referido anteriormente, o sistema de controlo proposto segue uma metodologia activa de controlo tolerante a falhas baseada em métodos projecção, que por definição assentam na determinação em diferido das diversas dinâmicas que o processo pode apresentar em funcionamento nominal ou com falhas.

Para as  $n$  dinâmicas nominais consideradas é proposto o dimensionamento de  $n$  parâmetros de controlo adequados, que são integrados na estrutura denominada *Base de Conhecimento*. Para um número  $m$  de falhas previstas são projectados  $m$  parâmetros de controlo que, em conjunto com os  $n$  parâmetros dimensionados para o funcionamento nominal, constituem a *Base de Conhecimento*, contendo assim  $m + n$  parâmetros de controlo.

A decisão dos parâmetros de controlo mais apropriados ao estado actual do sistema é realizada através de um conjunto de regras pré-definidas, que no seu conjunto compõem a estrutura denominada *Decisor*, ilustrado na figura 3.5.

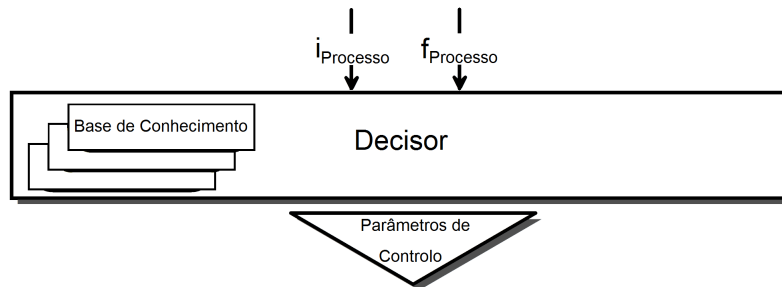


Figura 3.5: Decisor

Na ausência de falhas no processo, o *Decisor* terá de inferir através do valor  $Y_s$  obtido para a saída do sistema  $Y_p$ , quais os parâmetros a aplicar ao controlador. Deste modo, assumindo  $n$  intervalos diferentes para o funcionamento nominal do processo, o pseudo-código da estrutura *Decisor* é dado por:

#### Se funcionamento nominal

Se  $0 < Y_s < a$  então parâmetros de controlo = 1

Se  $a < Y_s < b$  então parâmetros de controlo = 2

Se  $b < Y_s < c$  então parâmetros de controlo = 3

...

Se  $x < Y_s < 1$  então parâmetros de controlo =  $n$

Para o caso de falha no processo o *Decisor* deverá, com base na identificação da falha fornecida pelo sistema de FDD, decidir quais os parâmetros de controlo que permitem a acomodação da falha. Deste modo, considerando que a falha no processo se encontra dentro do conjunto das  $m$  falhas previstas, o pseudo-código da estrutura *Decisor* é dado por:

**Se falha no processo**

Se  $f_{Processo} = 1$  então parâmetros de controlo =  $n + 1$

Se  $f_{Processo} = 2$  então parâmetros de controlo =  $n + 2$

Se  $f_{Processo} = 3$  então parâmetros de controlo =  $n + 3$

...

Se  $f_{Processo} = m$  então parâmetros de controlo =  $n + m$

Para o caso da falha se encontrar fora do conjunto  $m$  previsto ou ocorrer no nível de supervisão, impossibilitando o fluxo de informação necessária (detecção e identificação de falhas, correcções nos valores ou actuadores, sinais de referência etc.), existem diversas metodologias de acomodação. Algumas das soluções possíveis são a definição de uma referência / acção de controlo de segurança ou a paragem de emergência do sistema.

A implementação em Texto Estruturado da estrutura *Decisor* é ilustrada na figura 3.6. Para o exemplo ilustrado foram considerados quatro intervalos distintos de funcionamento nominal e quatro falhas previstas.

```
(* ----- Decisor -----*)
IF falha_processo=0 THEN
  (* Se Funcionamento nominal... *)
  IF y>0.00 AND y<=0.25 THEN param_controlo:=BC[1];
  ELSIF y>0.25 AND y<=0.50 THEN param_controlo:=BC[2];
  ELSIF y>0.50 AND y<=0.75 THEN param_controlo:=BC[3];
  ELSIF y>0.75 AND y<=1.00 THEN param_controlo:=BC[4]; END_IF;
ELSE
  (* Se Falha no processo... *)
  CASE falha_processo OF
    1: param_controlo:=BC[5];
    2: param_controlo:=BC[6];
    3: param_controlo:=BC[7];
    4: param_controlo:=BC[8];
  ELSE
    (*Caso a falha se encontre fora do conjunto previsto... *)
    ;
  END_CASE;
END_IF;
```

Figura 3.6: Decisor - Implementação em ST

A representação em FBD do código implementado resulta no bloco ilustrado na figura 3.7. Este bloco apresenta dois *inputs* - *Base de Conhecimento* e informação do supervisor - e três *outputs* - parâmetros de controlo, informação destinada ao sensor virtual e informação destinada ao actuador virtual.

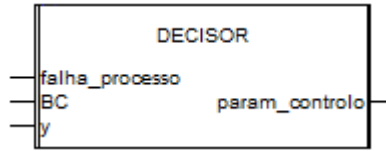


Figura 3.7: Decisor - Implementação em FBD

### 3.2.2 Sensores e Actuadores Virtuais

A acomodação de falhas nos sensores e actuadores é realizada através dos componentes *Sensores Virtuais* e *Actuadores Virtuais*, ilustrados nas figuras 3.9 e 3.8, respectivamente. Na presença de falhas nos sensores, o elemento *Sensor Virtual* corrige o valor  $Y_s$  para que este corresponda à saída real do processo  $Y_p$ . Esta correcção é efectuada através da adição do valor  $d_{Sensores}$ , dado pela diferença entre  $Y_p$  e  $Y_s$ . De forma análoga, o elemento *Actuador Virtual* realiza a correcção do valor da saída de actuação através da adição do valor  $d_{Actuadores}$ , dado pela diferença entre  $U_a$  e o  $U_c$ .

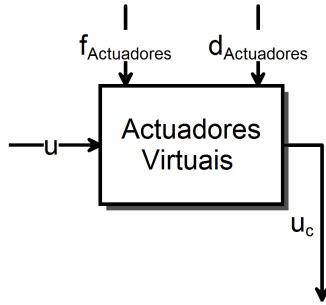


Figura 3.8: Actuador Virtual

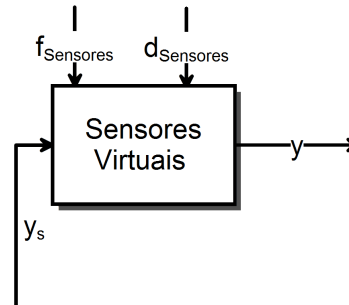


Figura 3.9: Sensor Virtual

O valor dos desvios  $d_{Sensores}$  e  $d_{Actuadores}$ , bem como a detecção de falha  $f_{Sensores}$  e  $f_{Actuadores}$ , é fornecido pelo nível de supervisão, mais concretamente pelo sistema de FDD. Sendo a detecção de falhas realizada através de modelos, é previsível que os valores reais se desviem com frequência dos valores previstos pelo sistema de FDD. Deste modo, a correcção contínua não é desejável uma vez o desvio por resultar da diferença entre o modelo e o processo e não de falha nos componentes. Assim, os elementos *Sensor Virtual* e *Actuador Virtual* só entram em funcionamento em caso de falha, sendo a sua existência transparente em funcionamento nominal. O pseudo-código para as estruturas descritas é dado por:

**Se falha no sensor**

$$Y = Y_s + d_{Sensores}$$

**Senão:**

$$Y = Y_s$$

**Se falha no actuador**

$$U_c = U + d_{Actuadores}$$

**Senão:**

$$U_c = U$$

A implementação em Texto Estruturado é ilustrada nas figuras 3.10 e 3.11. O encapsolamento em FDB das estruturas descritas resulta nos blocos ilustrados nas figuras 3.12 e 3.13.

```

(* ----- Actuador Virtual ----- *)
IF falha_actuador=TRUE THEN
  (* Se falha no actuador...*)
  uc:=u+dActuador;
ELSE
  (* Se funcionamento nominal...*)
  uc:=u;
END_IF;

```

Figura 3.10: Actuador Virtual - Implementação em ST

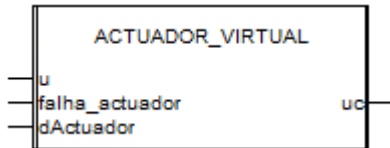


Figura 3.12: Actuador Virtual - Implementação em FBD

```

(* ----- Sensor Virtual ----- *)
IF falha_sensor=TRUE THEN
  (* Se falha no sensor...*)
  y:=ys+dSensor;
ELSE
  (* Se funcionamento nominal...*)
  y:=ys;
END_IF;

```

Figura 3.11: Sensor Virtual - Implementação em ST



Figura 3.13: Sensor Virtual - Implementação em FBD

### 3.2.3 Controlador

Uma das metodologias de controlo mais estudadas no mundo académico e amplamente utilizada em ambiente industrial é a metodologia *Proporcional Integral e Derivativa* (PID). Devido ao seu funcionamento simples e características robustas, esta abordagem permite resolver a grande maioria dos desafios de controlo para processos com dinâmicas de complexidade reduzida, sendo por isso a metodologia adoptada para a realização do sistema de controlo proposto nesta dissertação.

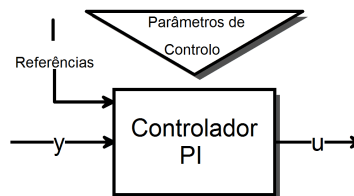


Figura 3.14: Controlador.

O princípio de funcionamento desta estrutura resulta da acção conjunta de três componentes - proporcional, integral e derivativa. A primeira componente fornece uma acção de controlo proporcional à diferença momentânea entre o valor real da saída do processo e a trajectória desejada. A componente proporcional é dada pela equação 3.1.

$$P(k) = K_p \cdot e(k) \quad (3.1)$$

$K_p$ : ganho proporcional;  $e(k)$ : erro de controlo no instante  $k$ .

Frequentemente, verifica-se que em regime estacionário a acção de controlo proporcional não é suficiente para levar a saída do processo até ao valor requerido. Com a introdução da componente

integral este problema pode ser resolvido uma vez que este elemento fornece uma acção de controlo proporcional ao erro de controlo acumulado. Deste modo, mesmo para erros de controlo reduzidos, a componente integral assegura uma acção de controlo crescente, aproximando a saída do sistema da referência pretendida. Esta componente é dada pela equação 3.2.

$$I(k) = I(k-1) + T_a \cdot K_p T_i \cdot e(k) \quad (3.2)$$

$K_p$ : ganho proporcional;  $T_i$ : tempo integral;  $e(k)$ : erro de controlo no instante  $k$ ;  $T_a$ : tempo de amostragem

A componente derivativa gera uma acção de controlo proporcional á predição do erro de controlo, sendo essencialmente utilizada para impedir regimes oscilatórios. No entanto, o dimensionamento desta componente é por vezes complexo e a sua utilização limitada a determinados casos. Na maioria das situações a componente derivativa não é utilizada sendo as componentes proporcional e integral suficientes para garantir um sistema de controlo eficaz. Deste modo, para o sistema de controlo proposto considera-se uma estrutura PI. Através das equações 3.1 e 3.2 obtém-se a acção de controlo  $u(k)$  dada pela equação 3.3.

$$u(k) = I(k-1) + K_p \cdot (1 + T_a T_i) \cdot e(k) \quad (3.3)$$

$K_p$ : Ganho proporcional;  $T_i$ : Ganho integral;  $e(k)$ : Erro de controlo no instante  $k$ ;  $T_a$ :Tempo de amostragem

### **Windup**

Um dos aspectos a considerar na implementação de sistemas de controlo é o facto de todos os actuadores reais terem a sua gama de acção limitada. Numa situação de saturação do actuador, a acção de controlo torna-se independente da saída do processo, permanecendo inalterada até que o sinal de controlo volta à gama de funcionamento nominal do actuador, introduzindo não linearidades que podem causar efeitos indesejados.

Em controladores que utilizem componentes integrais, o facto de a acção de controlo permanecer inalterada e independente da saída do processo constitui um problema. Considere-se o caso de saturação do actuador no seu limite máximo. Nesta situação o erro de controlo continuará a ser integrado aumentando o valor da componente integral para níveis anormalmente elevados, sendo necessários longos períodos de erro de controlo negativo para que normalize. Deste modo é introduzido um atraso entre a adaptação da acção de controlo e a saída do sistema, podendo este atraso conduzir o sistema à instabilidade.

Este fenómeno de saturação do actuador e consequente saturação do integrador, ilustrado na figura

3.15, é conhecido como efeito de *Windup* e existem diversos métodos estudados para impedir o seu aparecimento.

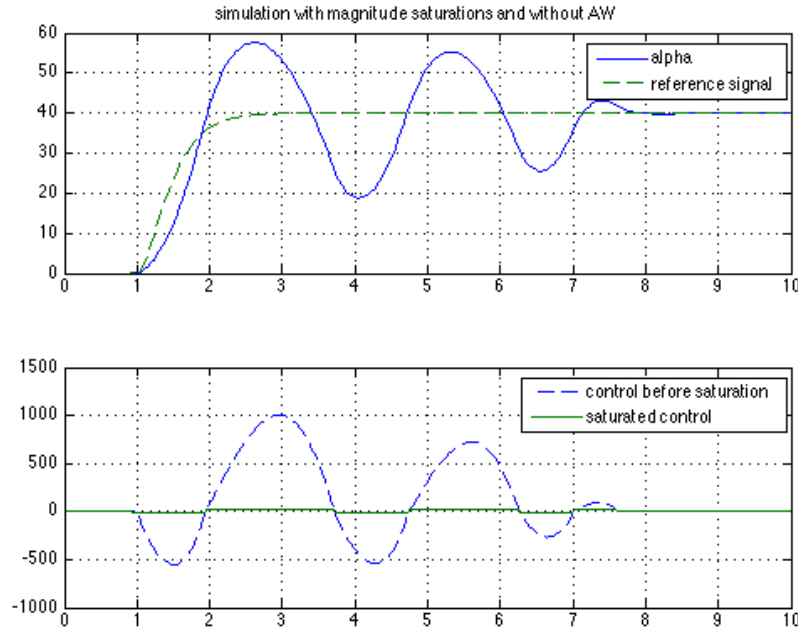


Figura 3.15: Exemplo de *Windup*

Um método estudado para evitar o fenómeno de *Windup* é a integração condicional. Este método propõe a suspensão da componente integral sempre que o sistema se encontre em regimes transitórios, sendo a componente integral apenas utilizada quando determinadas condições estão reunidas. As condições de utilização da componente integral podem ser definidas de diversas formas, sendo a mais comum a não integração do erro de controlo sempre que este é elevado ou durante a saturação da acção de controlo. No entanto, estas duas condições não são suficientes para garantir o bom desempenho do sistema, uma vez que podem resultar em comportamentos oscilatórios se o erro de controlo for elevado no momento em que a integração é suspensa.

De forma a evitar este problema é necessário considerar a condição adicional de apenas se proceder à suspensão da componente integral se a actualização desta resultar numa acção de controlo ainda mais saturada. O pseudo-código para o método de *Anti-Windup* descrito é dado por:

**Se actuador saturado no limite máximo e  $I_n > 0$**

$$I = I_o$$

**Se actuador saturado no limite mínimo e  $I_n < 0$**

$$I = I_o$$

**Se actuador dentro da gama de funcionamento**

$$I = I_n$$

$I_n$ : componente integral no instante  $k$ ;  $I_o$ : componente integral no instante  $k - 1$ .

A implementação através de Texto Estruturado do pseudo-código anterior é ilustrado na figura 3.16.

```
(* Mecanismo de Anti-Windup *)
IF ((P+In)>1.0 AND In>0.0) OR ((P+In)<0.0 AND In<0.0) THEN
    (* Se actuador saturado... *)
    u:=P+Io;
ELSE
    (* Se actuador não saturado... *)
    u:=P+In;
    Io:=In;
END_IF;
```

Figura 3.16: *Anti-Windup* - Implementação em ST

## Comutação

A alteração de um ou vários componentes de um controlador conduz inevitavelmente à alteração da sua dinâmica e consequentemente à alteração do seu sinal de saída. A alteração repentina do sinal de controlo pode trazer consequências indesejadas, devendo este efeito ser considerado e atenuado.

Para que o sistema de controlo se mantenha estável é necessário garantir um intervalo de tempo mínimo para comutações consecutivas, denominado dwell-time (Liberzone, 2003). Adicionalmente, para assegurar uma alteração de parâmetros sem mudanças repentinas do sinal de controlo, é necessário proceder-se á alteração dos parâmetros de forma a que o sinal de controlo permaneça constante no momento da comutação. Esta condição traduz-se na equação 3.4.

$$P_o + I_o = P_n + I_n \quad (3.4)$$

$P_o$ :componente proporcional instante  $k - 1$ ;  $I_o$ :componente integral no instante  $k - 1$ .

$P_n$ :componente proporcional no instante  $k$ ;  $I_n$ :componente integral no instante  $k$ .

Substituindo na equação anterior os valores de  $P_o$  e  $P_n$  através da aplicação da equação 3.1 obtém-se a equação 3.5

$$K_{po} \cdot e(k) + I_o = K_{pn} \cdot e(k) + I_n \quad (3.5)$$

$K_{po}$ :ganho proporcional instante  $k - 1$ ;  $K_{pn}$ :ganho proporcional no instante  $k$ ;

Resolvendo a equação anterior em ordem a  $I_n$  obtém-se

$$I_n = I_o + (K_{po} - K_{pn}) \cdot e(k) \quad (3.6)$$



Deste modo, para se efectuar a alteração dos parâmetros e garantir uma acção de controlo sem oscilações no instante da comutação, é necessário que a componente integral se obtenha de acordo com a equação 3.6. A implementação em Texto Estruturado do elemento de comutação descrito é ilustrada na figura 3.17.

```
(* Mecanismo de comutação *)
IF Kp <> param controlo.Kp THEN
    (* Se alteração de parâmetros... *)
    In:=Io+(Kp-param controlo.Kp)*(r-y);
    (* Actualização de Kp *)
    Kp:=param controlo.Kp;
ELSE
    (* Se parâmetros constantes... *)
    In:=Io+(Kp/param controlo.Ti)*(r-y);
END_IF;
```

Figura 3.17: Comutação - Implementação em ST

A implementação em texto estruturado do algoritmo de controlo PI, com os mecanismos de *Anti-Windup* e de comutação descritos, é ilustrada na figura 3.18. O código implementado inicia-se com uma componente de inicialização das variáveis do controlador, cuja execução deve ser realizada apenas no início do programa de controlo. O restante código é executado de forma cíclica, sendo este aspecto relevante no cálculo do valor da componente integral. Deste modo, a acção integral calculada na presente iteração (*In*) é guardada na variável *Io* de forma a ser utilizada no cálculo desta componente na iteração seguinte.

```
(* ----- Controlador PI ----- *)

(* Inicialização de parâmetros internos *)
IF Init=TRUE THEN
    Io:=0.0;
    Kp:=param controlo.Kp;
    u:=0.0;
ELSE
    (* Mecanismo de comutação *)
    IF Kp <> param controlo.Kp THEN
        (* Se alteração de parâmetros... *)
        In:=Io+(Kp-param controlo.Kp)*(r-y);
        (* Actualização de Kp *)
        Kp:=param controlo.Kp;
    ELSE
        (* Se parâmetros constantes... *)
        In:=Io+(Kp/param controlo.Ti)*(r-y);
    END_IF;

    (* Actualização de P *)
    P:=Kp*(r-y);

    (* Mecanismo de Anti-Windup *)
    IF ((P+In)>1.0 AND In>0.0) OR ((P+In)<0.0 AND In<0.0) THEN
        (* Se actuador saturado... *)
        u:=P+Io;
    ELSE
        (* Se actuador não saturado... *)
        u:=P+In;
        Io:=In;
    END_IF;
END_IF;
```

Figura 3.18: Controlador - Implementação em ST

Na figura 3.19 são ilustrados os blocos FBD correspondentes às estruturas de controlo descritas neste capítulo, apresentando adicionalmente o respectivo código em Texto Estruturado. A execução dos blocos de controlo é determinada pela identificação apresentada no canto superior direito de cada bloco FBD.

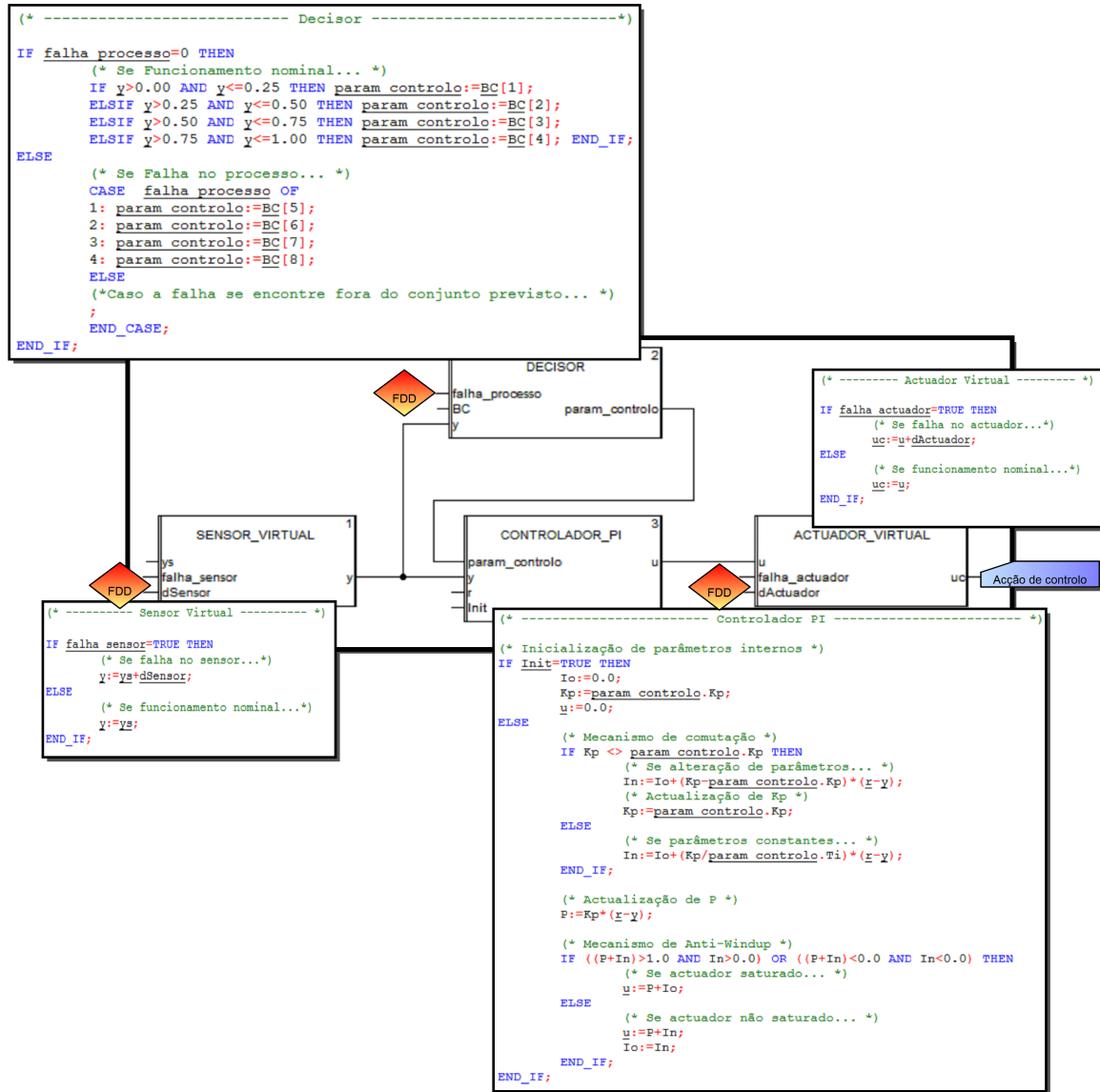


Figura 3.19: Nível de Controlo - Implementação em ST e FBD

## Capítulo 4

# Resultados Experimentais

Neste capítulo apresenta-se a metodologia de aplicação do sistema de controlo proposto ao processo laboratorial de dois tanques *FBK 38-100*, instalado no laboratório de automação da Faculdade de Ciência e Tecnologia. O processo em questão apresenta duas saídas mensuráveis, correspondentes aos níveis de água em cada tanque, e uma entrada de actuação que permite regular o caudal de circulação do sistema.

Os algoritmos de controlo e supervisão foram implementados em autómatos programáveis distintos, ambos do fabricante Schneider<sup>®</sup>, modelo Modicon M320, sendo a comunicação entre os dois dispositivos efectuada através da tecnologia *Ethernet*, utilizando o protocolo TCP/IP.

No final do capítulo apresentam-se os resultados obtidos dos ensaios realizados ao sistema, em funcionamento nominal e na presença de falhas, e conclui-se sobre o desempenho, viabilidade e condições de aplicação da proposta apresentada nesta dissertação.

## 4.1 Arquitectura

A aplicação ao caso de estudo considerado da metodologia estrutural proposta (apresentada na figura 3.1 do capítulo anterior) é ilustrada na figura 4.1. Para a implementação prática é necessário considerar interfaces entre os três níveis que possibilitam o fluxo de informação. Deste modo, entre os níveis de controlo e processo existe um conjunto de electrónica auxiliar, descrita no anexo A, necessária à troca de informação entre o dispositivo controlador e os elementos sensores e actuadores do processo. A comunicação entre os dispositivos de controlo e de supervisão é realizada através da tecnologia *Ethernet*, utilizando o protocolo TCP/IP.

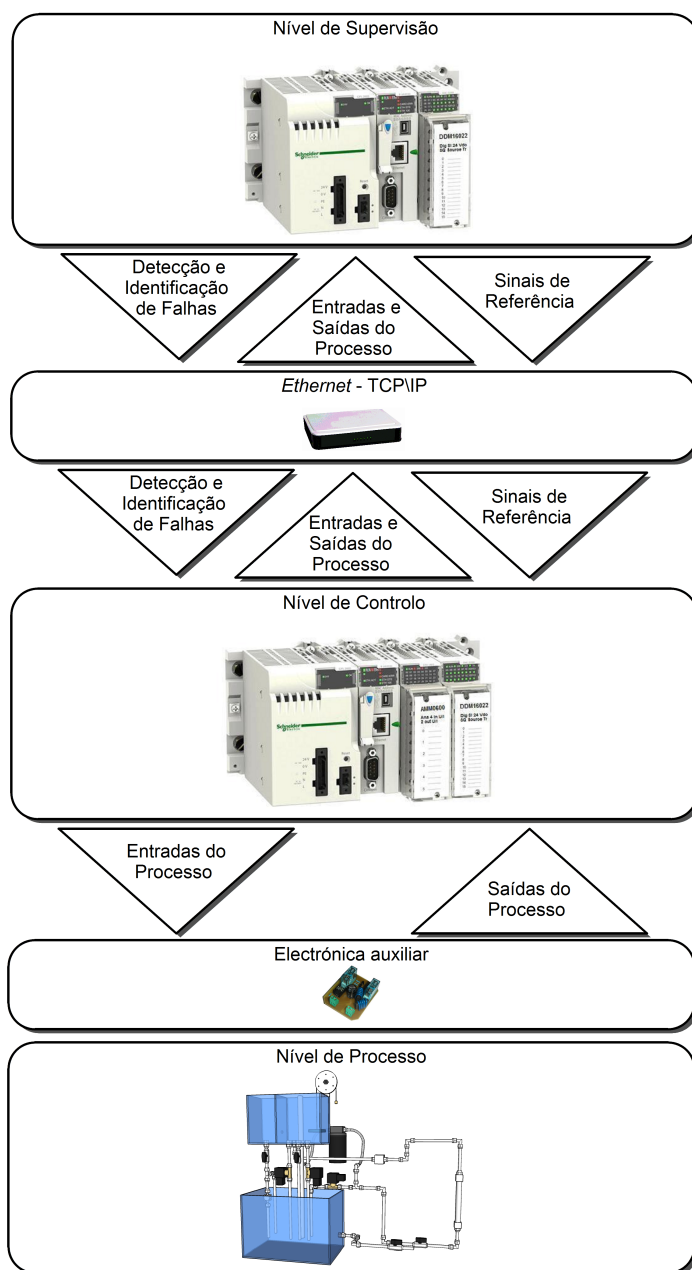


Figura 4.1: Arquitectura.

### 4.1.1 Nível de Processo

O nível de processo, localizado na base da arquitectura ilustrada na figura 4.1 é constituído pelo processo laboratorial, ilustrado na figura 4.2. Este sistema de circulação fechado é constituído por três tanques abertos - tanque 1, tanque 2 e tanque reservatório. Em funcionamento nominal, a circulação inicia-se pela passagem de água do tanque reservatório para o tanque 1, deste para o tanque 2, retornando ao tanque reservatório.

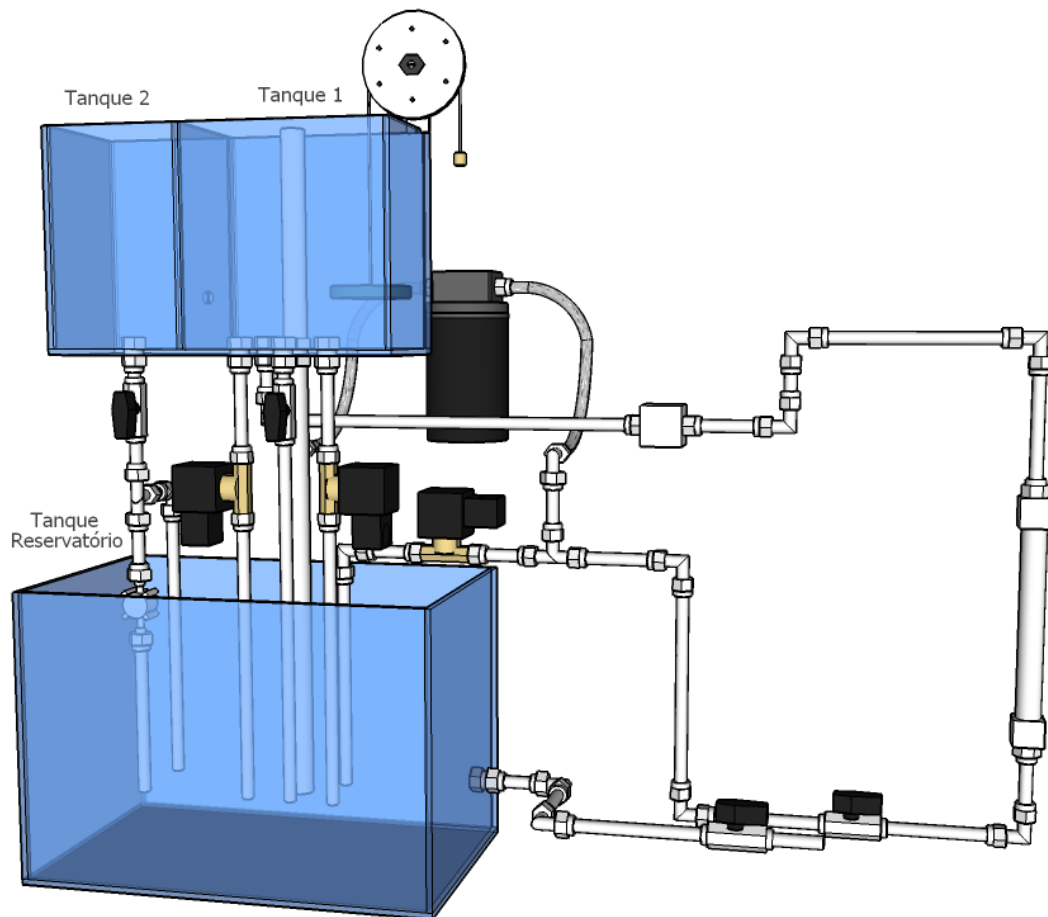


Figura 4.2: Processo laboratorial

A passagem de água do tanque reservatório para o tanque 1 é realizada através da acção de uma bomba de circulação, de caudal variável entre 0 e 4 litros p/ minuto. A regulação do caudal permite controlar directamente o nível do tanque 1 e, por se tratar de um sistema de vasos comunicantes, controlar indirectamente o nível do tanque 2. O fluxo entre os tanques 1 e 2 é realizado através de um orifício situado na face lateral adjacente aos dois reservatórios, o que implica a existência de diversas dinâmicas não lineares, descritas na segunda secção deste capítulo.

O nível de água do tanque 1 é aferido através do sensor de nível ilustrado na figura 4.3. Através do circuito eléctrico ilustrado no anexo, o deslocamento vertical do componente flutuador é transformado num sinal eléctrico contínuo, cujo valor é proporcional ao nível de água no tanque 1. Este tipo de sensores apresentam alguns erros de medição de origem mecânica e eléctrica, normalmente causados por

turbulência na água, atrito no deslocamento vertical do flutuador, atrito no deslocamento do elemento giratório, não linearidades nos componentes electrónicos e histerese do componente flutuador.

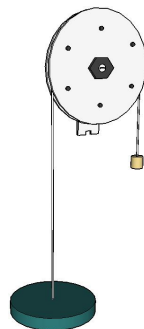


Figura 4.3: Sensor de nível - tanque 1

O nível do tanque 2 é obtido através da medição do caudal de passagem entre o tanque 2 e o tanque reservatório. A passagem de água actua sobre um componente giratório de elementos magnéticos, cuja velocidade de rotação é proporcional ao caudal. A alteração do campo magnético provoca o aparecimento de um sinal sinusoidal que, através do circuito electrónico ilustrado no anexo, gera um sinal eléctrico contínuo, cujo valor é proporcional à frequência da senoide e portanto proporcional ao nível do tanque 2. Os erros de medição que ocorrem neste sensor têm origem em não linearidades dos componentes electrónicos anexos e em impurezas que provoquem a obstrução do caudalímetro, ilustrado na figura 4.4.

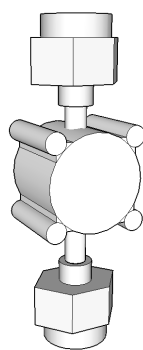


Figura 4.4: Sensor de caudal - tanque 2

Na base do tanque 1 estão instaladas duas electroválvulas -  $EV_1$  e  $EV_2$  - com diferentes diâmetros, que permitem que a água escoe do tanque 1 para o tanque o reservatório. Para além destas, existe a electroválvula -  $EV_3$  - localizada à saída da bomba permitindo que toda a água bombeada escoe directamente para o tanque reservatório, não sendo assim realizado o circuito nominal de circulação. Em funcionamento nominal, todas as electroválvulas encontram-se fechadas, sendo a abertura de qualquer uma das electrovalvulas considerada uma fuga no tanque 1, ou seja, uma falha no sistema. Os três componentes descritos, ilustrados na figura 4.5 são controlaods através do circuito electrónico auxiliar descrito no anexo.

Através da configuração dos canais de aquisição de dados do autómato, os sinais de saída do processo são processados por um filtro passa-baixo de 6ª ordem, de forma a atenuar o ruído eléctrico proveniente dos sensores e dos circuitos electrónicos auxiliares.

Em cada tanque estão instaladas válvulas manuais  $V_{m1}$ ,  $V_{m2}$  e  $V_{mr}$  que não serão consideradas para o caso de estudo, sendo apenas utilizadas para a manutenção do processo. Adicionalmente, encontra-se instalada na tubagem a válvula manual  $V_{m4}$  que permite a interrupção da circulação de água em todo o circuito. Todas as válvulas manuais são de uso restrito, devendo permanecer abertas apenas as válvulas  $V_{m2}$  e  $V_{m4}$  sempre que o sistema se encontre em funcionamento.

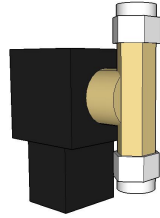


Figura 4.5: Electroválvula.

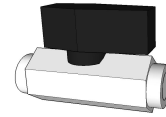


Figura 4.6: Válvula manual.

O diagrama funcional do processo é ilustrado na figura 4.7. Por não serem elementos relevantes na dinâmica do sistema, não foi considerado na ilustração o tanque reservatório e as válvulas manuais.

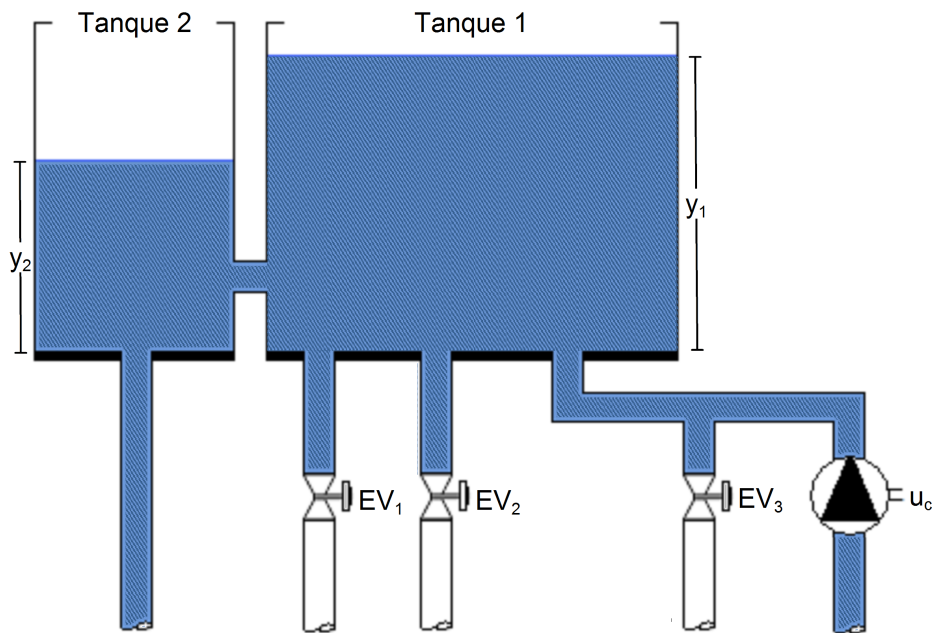


Figura 4.7: Processo laboratorial - Diagrama funcional

Em suma, o processo é constituído por uma entrada de actuação e duas saídas mensuráveis. A entrada de actuação  $u_c$  permite a regulação do caudal de entrada no tanque 1 e, consequentemente, a sua altura  $y_1$ . O nível do tanque 2,  $y_2$ , é determinado pelo fluxo entre tanques que, devido à localização do orifício de passagem, é caracterizado por diversas dinâmicas não lineares. As saídas  $y_1$  e  $y_2$  são obtidas através dos sensores de nível e de caudal, respectivamente. Estes dispositivos de aquisição são susceptíveis a falhas de origem eléctrica e mecânica que podem alterar o seu funcionamento nominal, e consequentemente, corromper os valores de  $y_1$  e  $y_2$ . Adicionalmente, foram instaladas no processo três electroválvulas,  $EV_1$ ,  $EV_2$ , e  $EV_3$  que permitem criar caudais de escoamento entre o tanque 1 e o tanque reservatório que, para a configuração considerada, são entendidos como fugas no tanque 1, ou seja, falhas no processo.

### 4.1.2 Nível de Controlo

O algoritmo de controlo descrito no capítulo 3 foi implementado no autómato ilustrado na figura 4.8. O autómato é constituído pelos módulos de alimentação, processamento, actuação e aquisição de dados, instalados no *rack* de 4 posições, modelo *BMX XBP 0400*.

1. **BMX CPS 2000:** Módulo de alimentação responsável pelo fornecimento energia a todos os dispositivos instalados no *rack* *BMX XBP 0400*.
2. **BMX P34 2030:** Módulo central de processamento no qual é executado o sistema de controlo. Este componente dispõe de uma porta de comunicação *Ethernet* através da qual se realiza o fluxo de dados com o autómato supervisor. Adicionalmente possui uma porta da tecnologia *CANOpen*, não utilizada nesta dissertação.
3. **BMX AMM 0600:** Módulo de aquisição e actuação analógico, com 4 *inputs*, dois deles utilizados para aquisição dos valores de  $y_1$  e  $y_2$ , e 2 *outputs*, sendo um dos canais utilizado para a regulação da bomba de circulação. Como foi referido anteriormente, o actuador e os sensores não se encontram directamente conectados ao autómato, necessitando da interface electrónica descrita nos anexos. As características técnicas deste módulo são descritas no anexo.
4. **BMX DDM 16022:** Módulo de aquisição e actuação digital com 8 *inputs* e 8 *outputs*. Este dispositivo foi instalado com vista a trabalhos futuros, não tendo aplicação neste caso de estudo concreto.

### 4.1.3 Nível de Supervisão

Os algoritmos de detecção e identificação de falhas, não descritos nesta dissertação, foram implementados no autómato ilustrado na figura 4.9, composto por módulos idênticos aos utilizados no autómato controlador.



Figura 4.8: Autómato controlador.



Figura 4.9: Autómato supervisor.



## 4.2 Metodologia

O objectivo principal do sistema de controlo consiste na regulação do nível de água do tanque 2, através da regulação do caudal de entrada. Este objectivo deve ser assegurado apesar da existência de perturbações ou falhas, que podem ocorrer nos sensores, corrompendo os valores medidos de  $y_1$  e  $y_2$ , nos actuadores, alterando a relação entre o sinal de controlo  $u_c$  e sinal de actuação  $u_a$ , ou ainda no processo, alterando a sua dinâmica de funcionamento.

O facto do fluxo de água entre os dois tanques se efectuar por uma passagem localizada na parede lateral a uma altura  $h_p$ , implica a necessidade de considerar várias dinâmicas de funcionamento que variam com os níveis dos tanques 1 e 2. Deste modo, o sistema apresenta quatros estados de funcionamento, representados nas figuras 4.10 a 4.13

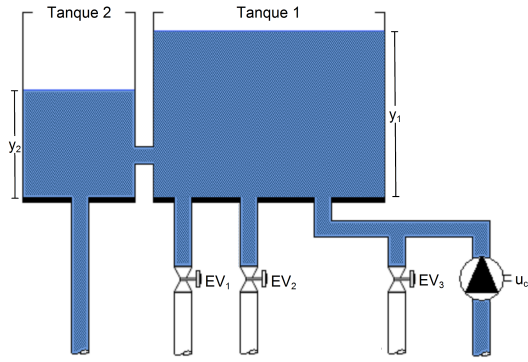


Figura 4.10: Estado 1:  $y_1 > h_p$  e  $y_2 > h_p$ .

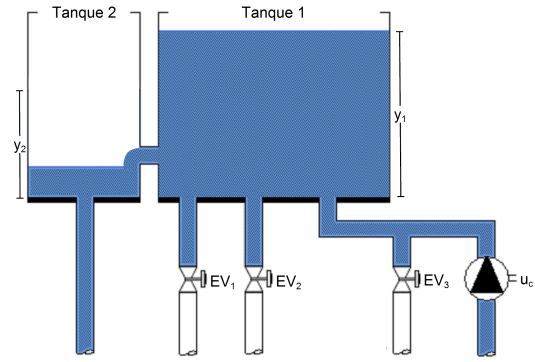


Figura 4.11: Estado 2:  $y_1 > h_p$  e  $y_2 < h_p$ .

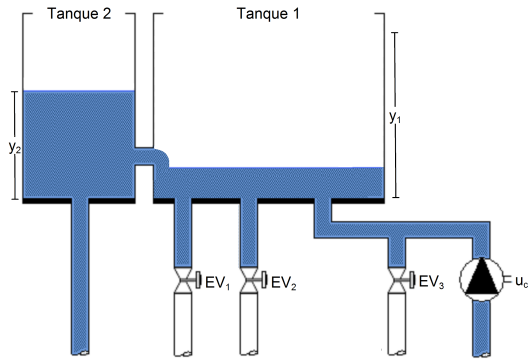


Figura 4.12: Estado 3:  $y_1 < h_p$  e  $y_2 > h_p$ .

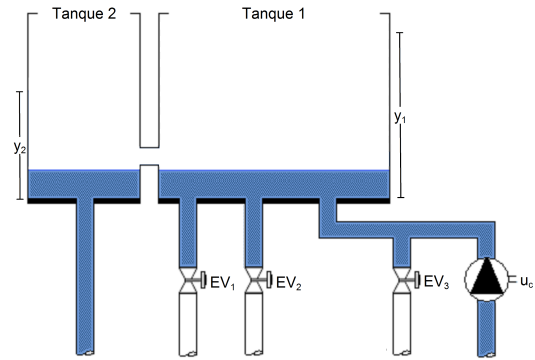


Figura 4.13: Estado 4:  $y_1 < h_p$  e  $y_2 < h_p$ .

A variação de volume nos tanques pode ser obtida através da lei da conservação de massa, dada pela equação 4.1.

$$\frac{dV}{dt} = A \frac{dh}{dt} = \sum Q_e - \sum Q_s \quad (4.1)$$

A: secção; h: altura; V: volume;  $Q_e$ : caudais de entrada;  $Q_s$ : caudais de saída;

Aplicando a equação 4.1 aos tanques 1 e 2 obtêm-se as equações 4.2 e 4.3, respectivamente.

$$\frac{dV_1}{dt} = A_1 \frac{dh_1}{dt} = q_e - q_{12} \quad (4.2)$$

$A_1$ : secção do tanque 1;  $h_1$ : nível do tanque 1;  $q_{12}$ : cauda entre tanques;  $q_e$ : caudal de entrada

$$\frac{dV_2}{dt} = A_2 \frac{dh_2}{dt} = q_{12} - q_s \quad (4.3)$$

$A_2$ : secção do tanque 2;  $h_2$ : nível do tanque 2;  $q_{12}$ : cauda entre tanques;  $q_s$ : caudal de saída

O caudal de entrada do tanque 1 ( $q_e$ ) corresponde ao caudal produzido pela bomba de água, cujo funcionamento é regulado pelo sinal de controlo  $u_c$ . A relação entre  $q_e$  e  $u_c$  pode ser aproximada por uma função linear de primeira ordem dada pela equação 4.4.

$$q_e = K_a \cdot u_c + b_a \quad (4.4)$$

$K_a, b_a$ : parâmetros do actuador;  $u_c$ : sinal de controlo.

O fluxo entre dois tanques pode ser obtido através da lei de *Toricelli*, dada pela equação 4.5.

$$Q_{12} = \alpha_{12} \text{sgn}(H_1 - H_2) \sqrt{2g|H_1 - H_2|} \quad (4.5)$$

$K_a, b_a$ : parâmetros do actuador;  $u_c$ : sinal de controlo.

Através da aplicação da equação 4.5 para as diferentes situações ilustradas nas figuras 4.10 a 4.13, obtêm-se as equações 4.6 a 4.9, que traduzem o valor do fluxo entre os dois tanques para cada um dos quatro estados nominais.

$$\textbf{Estado 1: } y_1 > h_p \text{ e } y_2 > h_p \quad q_{12} = k_{12} \cdot \text{sgn}(h_1 - h_2) \cdot \sqrt{|h_1 - h_2|} \quad (4.6)$$

$$\textbf{Estado 2: } y_1 > h_p \text{ e } y_2 < h_p \quad q_{12} = k_{12} \cdot \sqrt{|h_1 - h_p|} \quad (4.7)$$

**Estado 3:**  $y_1 < h_p$  e  $y_2 > h_p$

$$q_{12} = -k_{12} \cdot \sqrt{|h_2 - h_p|} \quad (4.8)$$

**Estado 4:**  $y_1 < h_p$  e  $y_2 < h_p$

$$q_{12} = 0 \quad (4.9)$$

Como foi referido anteriormente, as falhas no processo ocorrem devido à abertura singular ou combinada das electroválvulas  $EV_1$ ,  $EV_2$  e  $EV_3$ . Deste modo, para cada um dos casos, ilustrados nas figuras 4.14 a 4.17, é necessário considerar mais quatro dinâmicas de funcionamento do sistema.

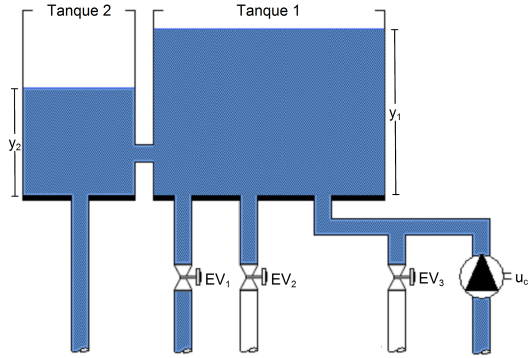


Figura 4.14: Falha 1: Abertura de  $EV_1$ .

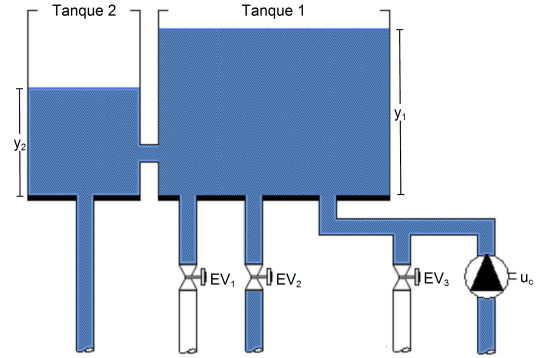


Figura 4.15: Falha 2: Abertura de  $EV_2$ .

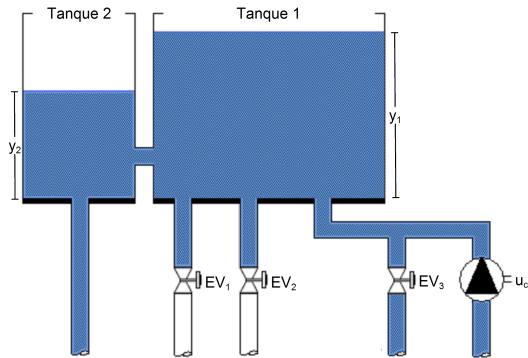


Figura 4.16: Falha 4: Abertura de  $EV_3$ .

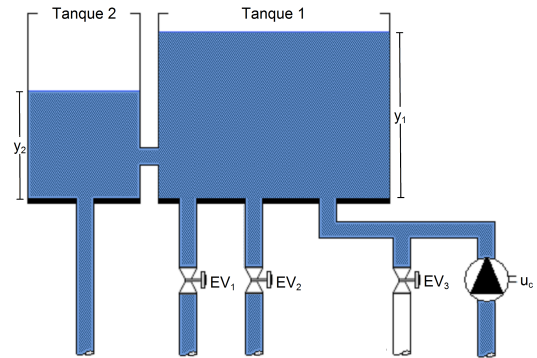


Figura 4.17: Falha 3: Abertura de  $EV_1$  e  $EV_2$ .

A abertura das electroválvulas  $EV_1$ ,  $EV_2$ ,  $EV_1 + EV_2$  e  $EV_3$  introduz uma fuga no tanque 1 de valor indeterminado e representada por  $q_{f1}$ ,  $q_{f2}$ ,  $q_{f12}$  e  $q_{f3}$ , respectivamente, com  $q_{f1} < q_{f2} < q_{f12} < q_{f3}$ . Através da lei de conservação de massa, obtém-se a equação 4.10, que traduz a dinâmica do tanque 1 em caso de falha.

$$\frac{dV_1}{dt} = A_1 \frac{dh_1}{dt} = q_e - q_{12} - Q_f \quad (4.10)$$

$Q_f$ : caudal resultante da falha  $\in \{ q_{f1}; q_{f2}; q_{f12}; q_{f3} \}$

Através da análise teórica da dinâmica do sistema conclui-se que o nível  $y_1$  e a sua variação dependem de  $y_2$  e do caudal de entrada  $q_e$ . No entanto, considerando as características físicas do sistema (a altura dos tanques e a secção do orifício de interligação entre eles) é possível fazer a aproximação ilustrada na figura 4.18. Deste modo, para o projecto de controlo foram consideradas duas sub-dinâmicas distintas. A dinâmica do tanque 1, onde  $y_1$  depende apenas do caudal de entrada de entrada  $q_i$ , e a dinâmica do tanque 2, onde  $y_2$  depende apenas de  $y_1$ .

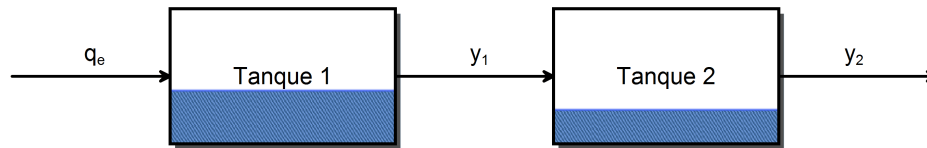


Figura 4.18: Dinâmica do sistema - aproximação considerada.

O controlo em cascata pode ser utilizado quando existem diversos sinais e apenas uma variável de controlo. É particularmente benéfico quando existem dinâmicas que apresentem, por exemplo, constante de tempo elevadas entre a acção de controlo e a saída do processo. Nestes casos, um controlo mais eficaz pode ser obtido utilizando variáveis intermédias que apresentem uma resposta mais rápida ao sinal de controlo.

Para o caso de estudo em questão, o nível  $y_1$  apresenta uma resposta mais rápida ao caudal de entrada  $q_e$ . A utilização de  $y_1$  como variável intermédia de controlo permite diminuir o tempo de resposta da acção de controlo  $u_c$  a variações do sinal de saída do sistema  $y_2$ . A estrutura de controlo cascata adoptada é ilustrado na figura 4.19.

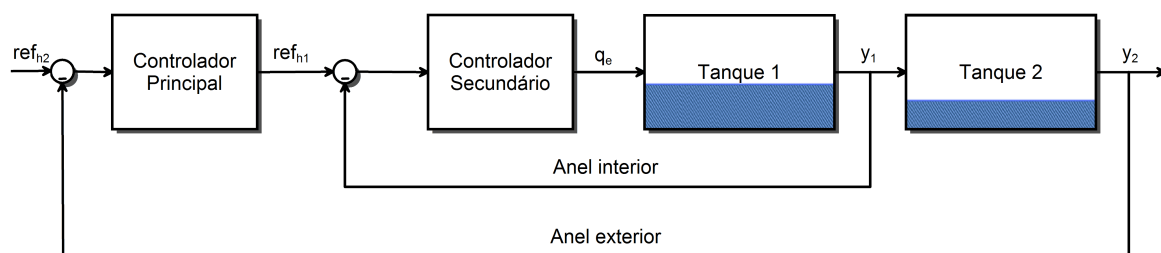


Figura 4.19: Diagrama de controlo em cascata.

A utilização de uma estrutura em cascata permite aumentar a robustez e o desempenho do sistema em situação de falha. Considere-se uma situação de falha, por exemplo a abertura da electroválvula  $EV_1$ , ilustrada na figura 4.14. Nesta situação, o nível  $y_1$  irá descer provocando o mesmo efeito, ao fim de algum tempo, no nível  $y_2$ . A manifestação da falha, primeiro no nível  $y_1$  e depois no nível  $y_2$ , permite que o controlador secundário gere uma acção correctiva, antes que a falha se manifeste no nível  $y_2$ , diminuindo assim o tempo de resposta à perturbação introduzida no sistema.

## 4.3 Implementação

Nesta secção será apresentada a implementação proposta para as estruturas de controlo referidas até aqui, utilizando o ambiente de programação Unity Pro®

### 4.3.1 Configuração de *Hardware*

O primeiro passo para a implementação do sistema de controlo no dispositivo ilustrado na figura 4.8, consiste na configuração do *hardware* utilizado. A definição dos módulos deve seguir a ordem pela qual estes estão dispostos no *rack*, como ilustrado na figura 4.20.

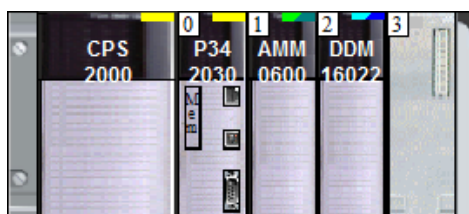


Figura 4.20: Módulo de processamento[0]; Módulo analógico[1]; Módulo digital[2]

### 4.3.2 Configuração de rede comunicação

A definição dos parâmetros de comunicação, necessários à troca de informação entre os dispositivos controlador e supervisor, deve ser realizada de acordo com características da rede já instalada. A cada um dos dispositivos deverá ser atribuído um endereço de IP fixo. Este ponto da implementação é ilustrado na figura 4.21

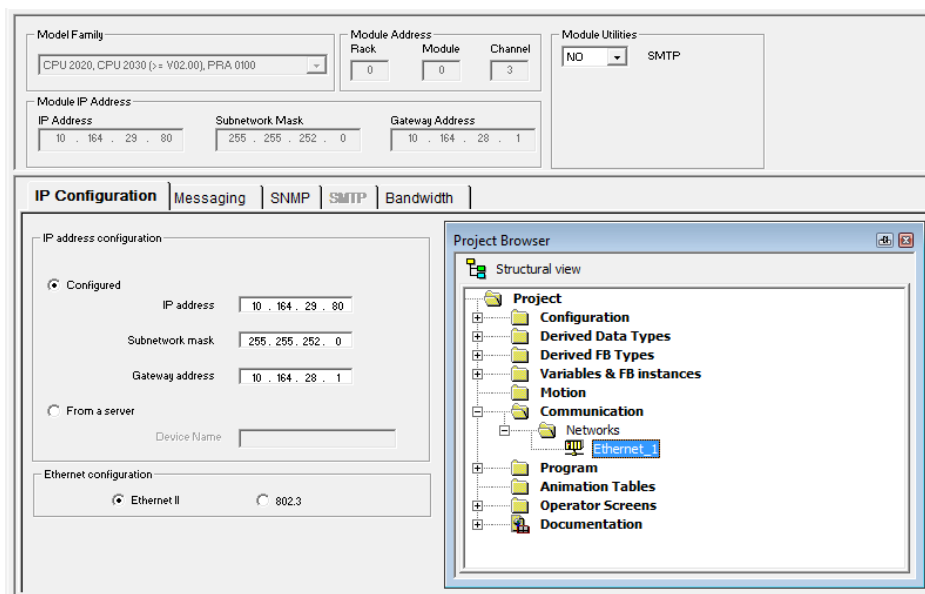


Figura 4.21: Definição dos parâmetros de rede

### 4.3.3 Definição de variáveis

Para a implementação do sistema de controlo é necessário proceder-se à declaração de variáveis. Nos pontos abaixo serão identificadas as variáveis para cada zona de funcionamento, nomeadamente variáveis destinadas à actuação e aquisição de dados, parâmetros de controlo e variáveis auxiliares à comunicação entre autómatos.

1. **Sensores e actuadores:** A definição das variáveis para aceder ao actuador e sensores conectados ao módulo analógico, segue a seguinte sintaxe:

*%Objecto\_Formato\_Rack.m.c*

**Objecto:** Indicação do tipo de objecto:

- I: Entrada analógica;
- Q: Saída analógica;
- M: Variável interna;
- K: Constante interna.

**Formato:** Indicação do formato do objecto:

- X: Boleano;
- W: *Single length (32 bits)*;
- D: *Double length (64 bits)*;
- F: *Floating point*.

**Rack:** Endereço do rack; **m:** Posição do módulo no rack; **c:** Canal ao qual se pretende aceder.

Na tabela 4.1 é apresentada a sintaxe relativa à definição das variáveis de entrada e saída do processo, de acordo com a configuração dos módulos ilustrada na figura 4.20 e de acordo com os canais aos quais foram conectados os sensores e actuadores.

Name	Type	Address	Value	Comment
I0	INT	%IW0.1.0		Input do módulo analógico (0).
I1	INT	%IW0.1.1		Input do módulo analógico (1).
O1	INT	%QW0.1.5		Output do módulo analógico (1).

Tabela 4.1: Variáveis de entrada e saída do módulo analógico.

2. **Tabela de controlo:** A tabela 4.2 contém às variáveis utilizadas no cálculo do sinal de controlo - referência para o nível 2, níveis dos tanques 1 e 2 e acções de controlo dos controladores principal e secundário. Todos os valores são normalizados entre 0 e 1.

Name	Type	Address	Value	Comment
Tabela_Controlo	ARRAY[0..4] OF REAL			Tabela de parâmetros do sistema de controlo
Tabela_Controlo[0]	REAL			Sinal de referência para o nível do tanque 2
Tabela_Controlo[1]	REAL			Nível do tanque 1.
Tabela_Controlo[2]	REAL			Nível do tanque 2.
Tabela_Controlo[3]	REAL			Sinal de controlo principal
Tabela_Controlo[4]	REAL			Sinal de controlo secundário

Tabela 4.2: Tabela de controlo.

3. **Parâmetros dos controladores:** A definição dos ganhos dos controladores PI projectados (determinados através de ajuste manual) é apresentada na tabela 4.4. Os controladores são constituídos por uma estrutura denominada *PARAM\_PI* que contem os parâmetros de controlo, ou seja, o ganho proporcional  $K_p$  e o tempo integral  $T_i$ .

Name	Type	Comment
PARAM_PI	<Struct>	
Kp	REAL	Ganho Proporcional
Ti	REAL	Tempo Integral

Tabela 4.3: Estrutura PARAM\_PI

Name	Type	Address	Value	Comment
Base_de_Conhecimento	ARRAY[0..4] OF PARAM_PI			Controladores secundários.
Base_de_Conhecimento[0]	PARAM_PI			Controlador secundário nominal.
Kp	REAL		3.35	Ganho proporcional
Ti	REAL		33.5	Tempo de Integração
Base_de_Conhecimento[1]	PARAM_PI			Controlador secundário - falha 1.
Kp	REAL		3.37	Ganho Proporcional
Ti	REAL		33.5	Tempo de Integração
Base_de_Conhecimento[2]	PARAM_PI			Controlador secundário - falha 2.
Kp	REAL		3.4	Ganho Proporcional
Ti	REAL		33.5	Tempo Integral
Base_de_Conhecimento[3]	PARAM_PI			Controlador secundário - falha 3.
Kp	REAL		3.45	Ganho Proporcional
Ti	REAL		33.5	Tempo Integral
Base_de_Conhecimento[4]	PARAM_PI			Falha irreversível
Kp	REAL		0.0	Ganho Proporcional
Ti	REAL		0.0	Tempo Integral
Controlador_Principal	PARAM_PI			Controladores principais.
Kp	REAL		1.5	Ganho Proporcional
Ti	REAL		37.5	Tempo Integral

Tabela 4.4: Parâmetros de controlo.

4. **Tabela de comunicação controlador → supervisor:** O fluxo de informação entre os sistemas de controlo e de supervisão é ilustrado na figura 4.22. As variáveis ilustradas na figura 4.5 são utilizadas para a comunicação no sentido controlador → supervisor. O controlador deverá fornecer ao supervisor os valores obtidos dos sensores e o valor colocado na saída de actuação. Para o supervisor aceder aos dados correctamente, é necessário definir a localização das variáveis, sendo essa informação um parâmetro essencial na transferência de dados entre autómatos.

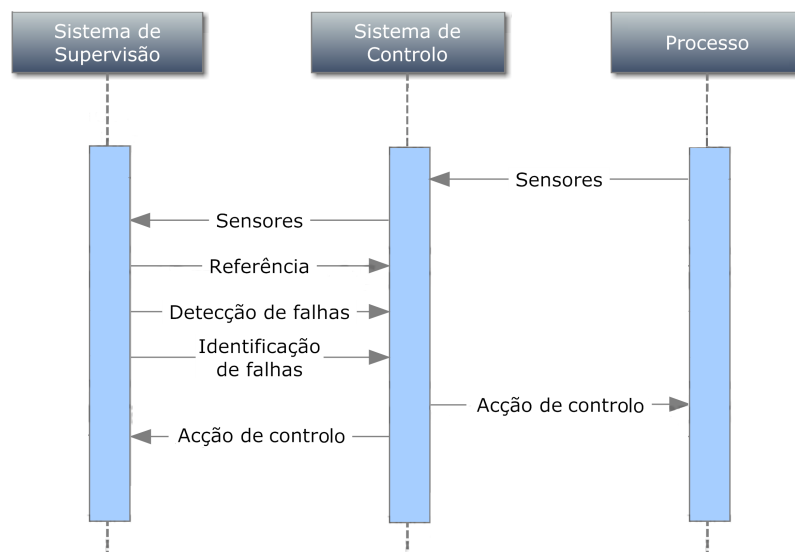


Figura 4.22: Diagrama de fluxo.

Name	Type	Address	Value	Comment
Tabela_Supervisor_Env	ARRAY[0..2] OF INT	%MW100		Tabela de parâmetros a enviar ao supervisor.
Tabela_Supervisor_Env[0]	INT	%MW100		Output do módulo analógico (1).
Tabela_Supervisor_Env[1]	INT	%MW101		Input do módulo analógico (0).
Tabela_Supervisor_Env[2]	INT	%MW102		Input do módulo analógico (1).

Tabela 4.5: Tabela de comunicação controlador → supervisor.

5. **Tabela de comunicação supervisor → controlador:** A tabela 4.6 contém as variáveis necessárias à comunicação no sentido supervisor → controlador. Em funcionamento nominal, o supervisor deverá fornecer ao controlador o valor de referência para o nível do tanque 2. Em caso de falha no actuador ou sensores, o supervisor deverá garantir a sua detecção e transmitir o valor dos desvios de  $y_1$ ,  $y_2$  e  $u_c$  em relação ao ponto de funcionamento nominal. Para o caso de falha no processo, o supervisor deverá transmitir a identificação da falha.

Name	Type	Address	Value	Comment
Tabela_Supervisor_Rcb	ARRAY[0..7] OF INT			Tabela de parâmetros a receber do supervisor.
Tabela_Supervisor_Rcb[0]	INT			Deteção de falha no actuador
Tabela_Supervisor_Rcb[1]	INT			Deteção de falha no sensor 1
Tabela_Supervisor_Rcb[2]	INT			Deteção de falha no sensor 2
Tabela_Supervisor_Rcb[3]	INT			Desvio do actuador
Tabela_Supervisor_Rcb[4]	INT			Desvio do sensor 1
Tabela_Supervisor_Rcb[5]	INT			Desvio do sensor 2
Tabela_Supervisor_Rcb[6]	INT			Identificação de falha no processo
Tabela_Supervisor_Rcb[7]	INT			Nível de referência para o tanque 2

Tabela 4.6: Tabela de comunicação supervisor → controlador.

Como se pode verificar nas tabelas de comunicação 4.5 e 4.6, todas as variáveis definidas são do tipo *INT*. Este facto deve-se à necessidade de todas as variáveis a transmitir ou a receber serem definidas desta forma. No entanto, o sistema de controlo utiliza variáveis do tipo *REAL*, como se pode verificar nas tabelas 4.2 e 4.4. De modo a compatibilizar o tipo de variáveis utilizadas, foram definidas as tabelas auxiliares 4.7 que contêm a informação relativa à detecção de falhas e os valores dos desvios do actuador e sensores, respectivamente.

Name	Type	Address	Value	Comment
Tabela_Correccao	ARRAY[0..2] OF REAL			Tabela de correcção de desvios
Tabela_Correccao[0]	REAL			Correcção do desvio no actuador
Tabela_Correccao[1]	REAL			Correcção do desvio no sensor 1
Tabela_Correccao[2]	REAL			Correcção do desvio no sensor 2
Tabela_Deteccao	ARRAY[0..2] OF BOOL			Tabela de detecção de falhas
Tabela_Deteccao[0]	BOOL			Deteção de falha no actuador
Tabela_Deteccao[1]	BOOL			Deteção de falha no sensor 1
Tabela_Deteccao[2]	BOOL			Deteção de falha no sensor 2

Tabela 4.7: Tabela auxiliar de detecção e correcção.



#### 4.3.4 Aquisição de dados

O primeiro passo para a execução do sistema de controlo é a aquisição dos dados do sistema, ou seja, determinar os níveis  $y_1$  e  $y_2$ . Na figura 4.23 são ilustrados os blocos em FBD responsáveis por este ponto.

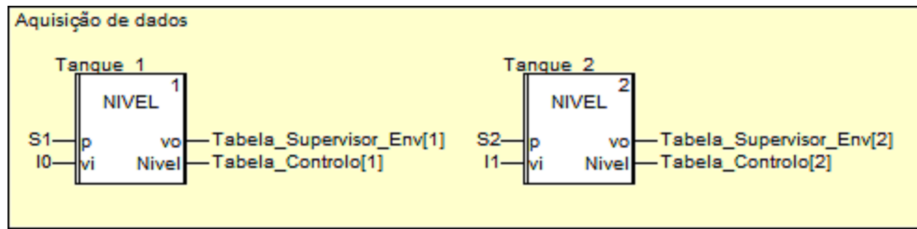


Figura 4.23: Aquisição de dados.

A aquisição de dados inicia-se pela leitura dos valores dos sensores, acedendo às variáveis da tabela 4.1. De seguida, procede-se à transformação do sinais obtidos pelos sensores nas grandezas  $y_1$  e  $y_2$ , através dos parâmetros de calibração previamente obtidos, S1 e S2, respectivamente. Após a obtenção de  $y_1$  e  $y_2$ , são actualizadas os valores das tabelas 4.2 e 4.5.

#### 4.3.5 Comunicação

A etapa da comunicação consiste na actualização das variáveis fornecidos pelo supervisor, declaradas na tabela 4.6. Este ponto é executado pelo bloco FBD ilustrado na figura 4.24.

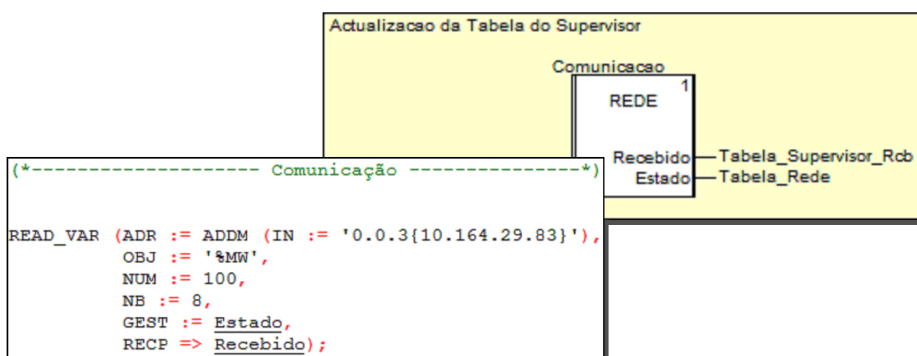


Figura 4.24: Comunicação.

O código em ST do bloco FBD de comunicação, ilustrado na figura 4.24, está disponível na biblioteca de funções do ambiente de programação Unity Pro<sup>®</sup>. Esta estrutura, denominada READ\_VAR, permite realizar a leitura de variáveis internas de outro dispositivo, neste caso do autómato supervisor. O bloco em questão apresenta os seguinte parâmetros de entrada:

- **ADR:** Este campo deverá conter o endereço de IP do dispositivo ao qual se pretende aceder, que deve ser precedido da indicação porta de *Ethernet* utilizada. Considerando a configuração

ilustrada na figura 4.20, a localização da porta é dada pela identificação do rack (0), posição do módulo de processamento (0), e pelo canal correspondente à porta de *Ethernet* (3);

- **OBJ:** Este parâmetro deverá conter o tipo dos objectos aos quais se pretende aceder, que para o caso questão são variáveis localizadas na memória interna do dispositivo supervisor, devendo por isso ser definido como %MW;
- **NUM:** Para se proceder à leitura de variáveis é necessário que estas tenham uma posição definida na memória interna do dispositivo ao qual se pretende aceder. Deste modo, no projecto do supervisor, a declaração das variáveis a transmitir deverá ser realizada de forma idêntica à descrita na tabela 4.5, sendo este um aspecto preponderante na sincronização e integração dos sistemas de controlo e supervisão. Este parâmetro deverá ser definido com o valor da posição de memória do primeiro objecto;
- **NB:** Este parâmetro de entrada define do número de objectos ao qual se pretende aceder devendo por isso ser igual ao total de variáveis declaradas na tabela 4.6;
- **GEST:** Este parâmetro deverá ser definido como uma tabela de cinco variáveis do tipo INT, que no final da execução da função READ\_VAR, irá conter a identificação do erro de comunicação, caso esta seja mal sucedida;
- **RECEP:** O valor das variáveis às quais se pretende aceder será guardado na estrutura passada por este parâmetro, que para o caso em questão é a tabela 4.6.

Como foi referido anteriormente, a transmissão de variáveis só é possível se estas forem definidas como sendo do tipo INT. De forma a que possam ser utilizadas pela estrutura de controlo, é necessário proceder-se à sua conversão para os tipos REAL e BOOL. Este ponto, ilustrado nas figuras 4.25 e 4.26, é realizado através dos blocos INT\_TO\_BOOL e INT\_TO\_REAL, disponíveis na biblioteca de funções do ambiente de programação. Após a conversão, os valores são guardados nas tabelas auxiliares 4.7.

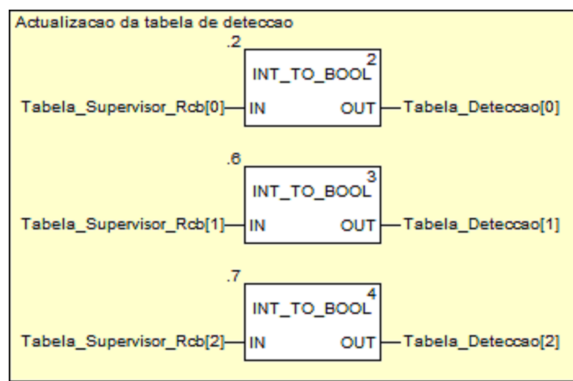


Figura 4.25: Conversão de variáveis de detecção.

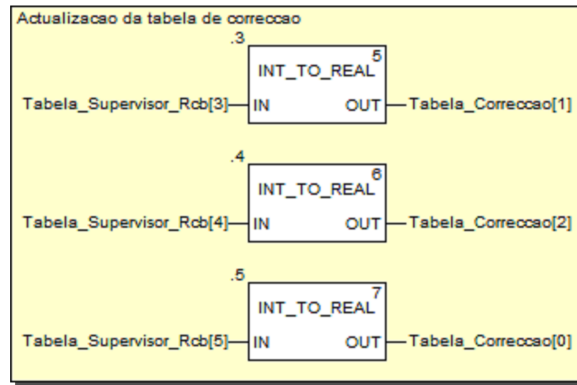


Figura 4.26: Conversão de variáveis de correcção.

#### 4.3.6 Estrutura de controlo

Após a aquisição de dados e a recepção da informação proveniente do nível de supervisão, procede-se ao cálculo da acção de controlo, efectuado pelas estruturas ilustradas na figura 4.27.

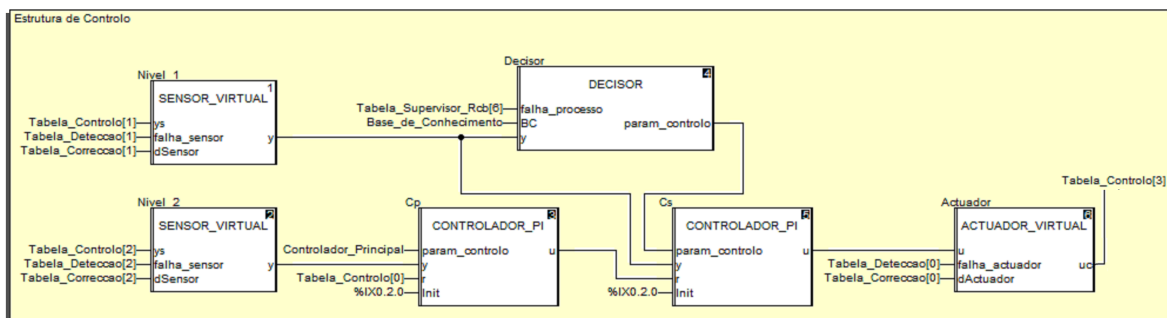


Figura 4.27: Estrutura de controlo - FBD.

A estrutura de controlo é constituída pelos sensores virtuais, controlador principal, controlador secundário, decisor e actuador virtual. Todos estes blocos FBD apresentam no canto superior direito a ordem pela qual são executados. Deste modo, o algoritmo de controlo inicia-se pela correcção dos valores  $y_1$  e  $y_2$ , caso se verifique falhas nos sensores. Posteriormente, é calculada a acção de controlo principal, que servirá de referência ao controlador secundário, estando esta limitada entre os valores normalizados 0 e 1. De seguida, o bloco decisor determina, com base na informação fornecida pelo nível de supervisão, quais os parâmetros de controlo adequados ao estado do sistema. Segue-se o cálculo da acção de controlo secundária, que será corrigida pelo actuador virtual, caso se verifique falha no actuador. Ao contrário das componentes de aquisição, comunicação e actuação, que são executados a taxa máxima de processamento do autómato, esta estrutura é executada de acordo com o tempo de amostragem do sistema de controlo, que foi definido como sendo de 1 segundo.

### 4.3.7 Actuação

O último passo para a execução do sistema de controlo consiste na determinação do valor de tensão correspondente ao caudal requerido. Este ponto é executado pelo bloco ACTUADOR (ilustrado na figura 4.28) que, através dos parâmetros de calibração A1, determina qual o valor de tensão correspondente à acção de controlo calculada. Posteriormente, é actualizada a saída de actuação e a tabela 4.5 com o valor de tensão obtido.

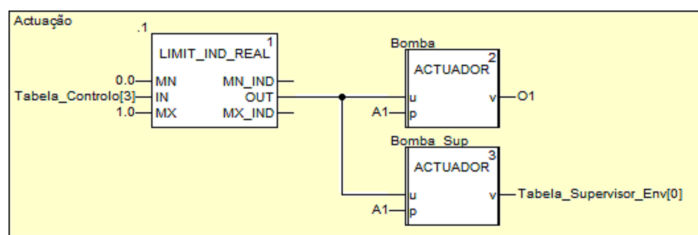


Figura 4.28: Actuação - FBD.

## 4.4 Resultados

Para testar o sistema de controlo proposto foram realizados testes em regime nominal e com falhas. Nos parágrafos seguintes apresentam-se os resultados obtidos.

### 4.4.1 Ensaio em Funcionamento Nominal

O ensaio ilustrado na figura 4.29 foi realizado com o objectivo de concluir sobre o comportamento do sistema de controlo em regime nominal, analisando tempos de subida, sobreelevação e erro estático. O ensaio foi iniciado com o nível do tanque 2 estabilizado nos 40%, sendo posteriormente introduzida uma referência igual a 60% durante 4 minutos. Após este periodo de tempo foi re-introduzida a referência de 40%, permanecendo inalterada até ao final do ensaio.

Com a alteração da referência do nível do tanque 2 para 60%, verifica-se a saturação momentânea do sinal de actuação, limitado a 90%. Este ponto permite aferir o bom funcionamento da metodologia de *anti-windup* adoptada, não permitindo o aparecimento dos efeitos de *windup* referidos no capítulo 3. Mostra também um tempo de subida de aproximadamente 60 segundos para uma variação positiva de 20%. De referir que, devido às dimensões dos tanques, o ajuste dos parâmetros de controlo foi realizado de forma a obter uma sobreelevação praticamente nula, sacrificando por isso o tempo de estabelecimento.

Devido às características do actuador, não é possível bombear água do tanque 1 para o tanque reservatório. Desta forma, a descida do nível de água é determinada pela secção do orifício de saída e pelo nível de água do tanque 2, provocando longos periodos de erro de controlo negativo, onde se

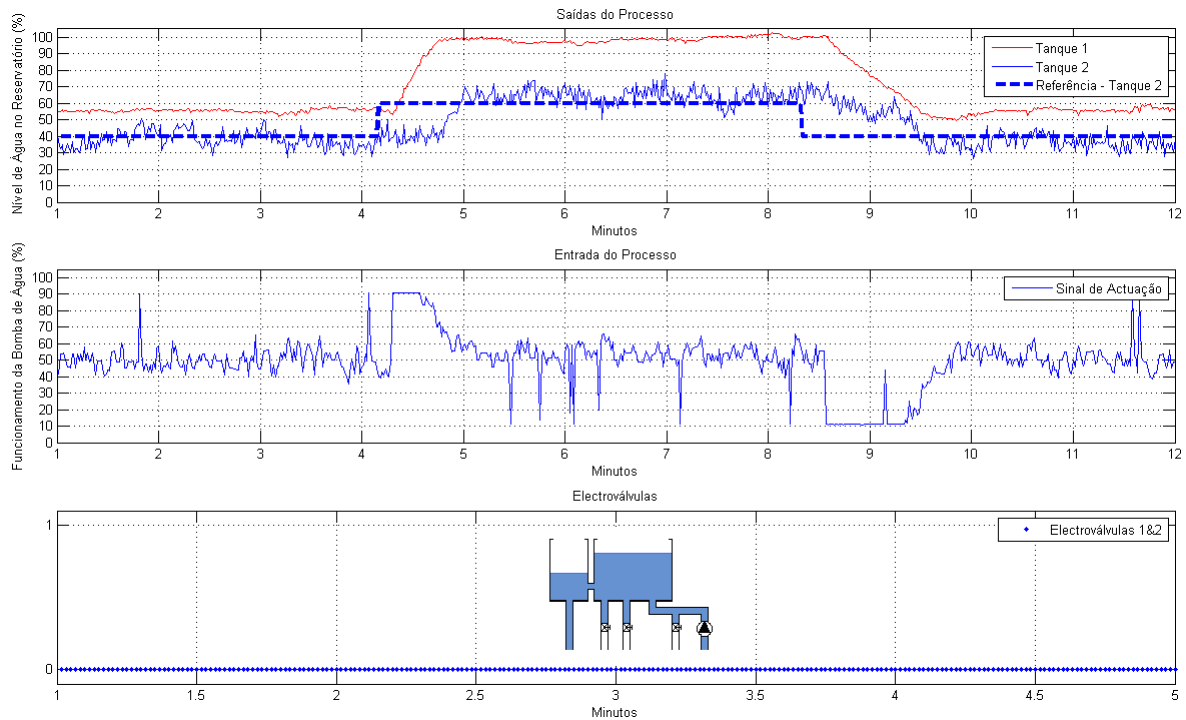


Figura 4.29: Ensaio nominal

verifica a saturação do actuador no seu limite mínimo de 10%. Também nesta situação, é possível verificar o bom desempenho do mecanismo de *Anti-Windup* adoptado.

Em suma, o sistema de controlo apresenta um comportamento satisfatório em termos de sobreelvação e de estabilidade. Devido às dimensões reduzidas dos tanques instalados é possível extrapolar estes resultados para outros pontos de funcionamento. Como é observável durante todo o ensaio, a medição do nível  $y_2$  é severamente corrompida por ruído, que, como referido anteriormente, tem origem nos componentes electrónicos complementares e nas impurezas existentes na água.

#### 4.4.2 Ensaio com Falhas

- **Falha 1:** Abertura da electroválvula  $EV_1$ , permitindo um caudal de escoamento  $q_{f1}$ , do tanque 1 para o tanque reservatório.
- **Falha 2:** Abertura da electroválvula  $EV_2$ , permitindo um caudal de escoamento  $q_{f2}$ , do tanque 1 para o tanque reservatório, com  $q_{f2} > q_{f1}$ .
- **Falha 3:** Abertura das electroválvulas  $EV_1$  e  $EV_2$ , permitindo um caudal de escoamento  $q_{f1} + q_{f2}$ , do tanque 1 para o tanque reservatório.

Nos três ensaios foi aplicada uma referência constante para o tanque dois igual a 50%. A abertura da electroválvula foi efectuada aos 3 minutos, permanecendo aberta até ao final do ensaio.

Os resultados obtidos nos ensaios 1 e 2, figuras 4.30 e 4.31, permitem concluir que o projecto de

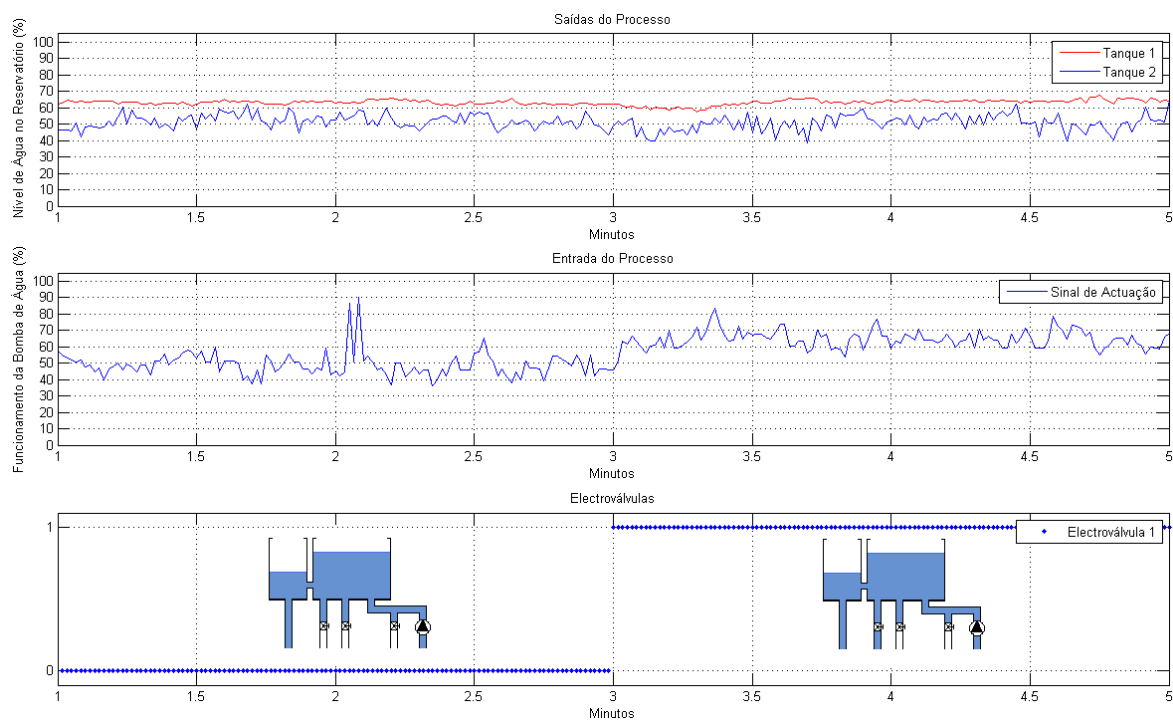


Figura 4.30: Ensaio - Falha 1

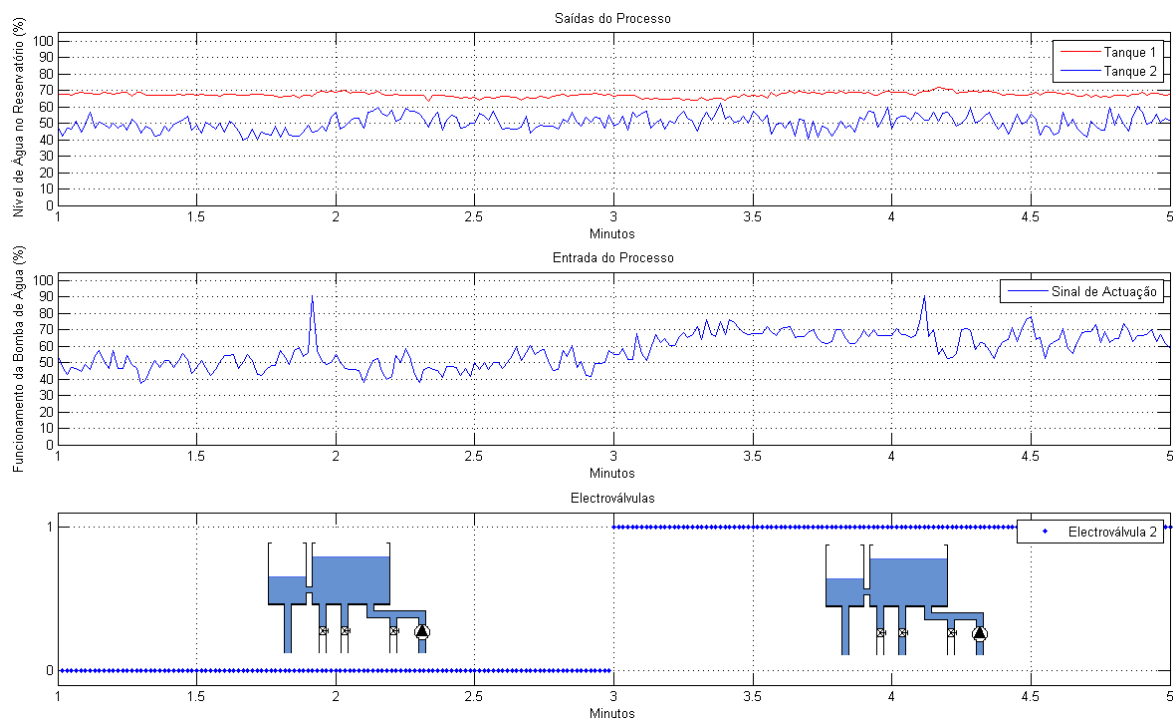


Figura 4.31: Ensaio - Falha 2

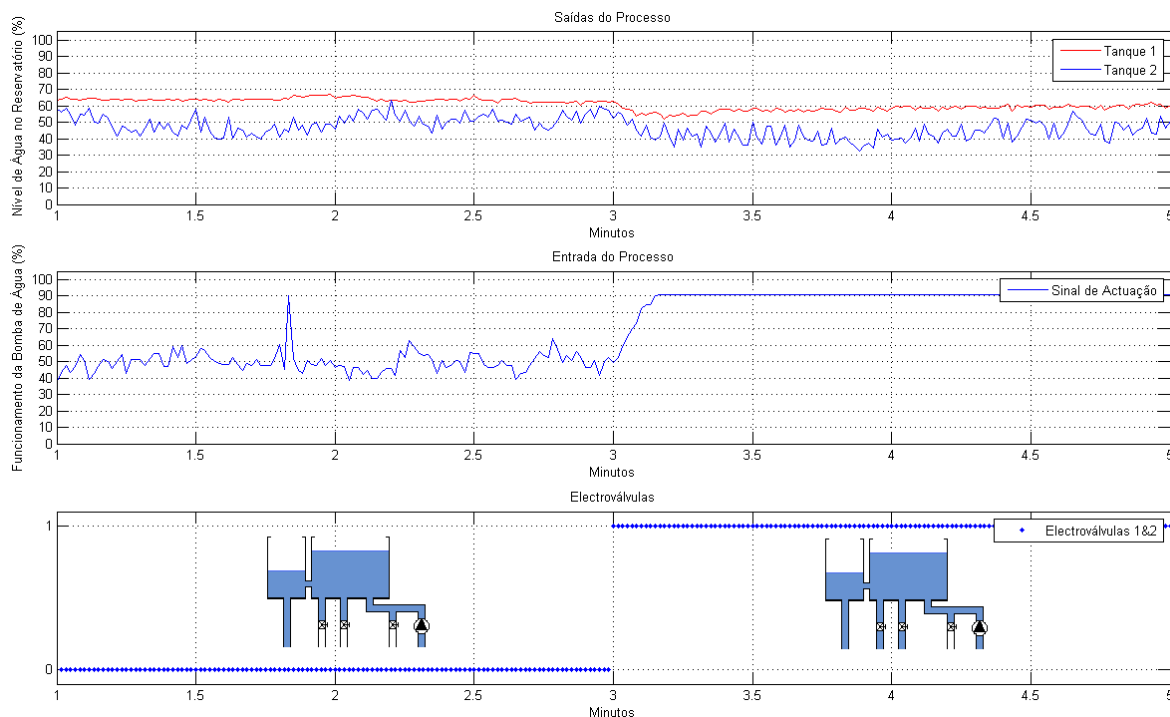


Figura 4.32: Ensaio - Falha 3

controlo mantém o desempenho do sistema na presença de falhas, adaptando em tempo útil a acção de controlo, de forma a compensar os caudais de fuga originados pela abertura das electroválvulas. O terceiro ensaio realizado, ilustrado na figura 4.32, provoca uma fuga no tanque 1 com um caudal superior ao necessário para manter o nível do tanque 2 a 50%, provocando a saturação da bomba de actuação.

Através da comparação dos ensaios 1 e 2, é possível observar que, para o mesmo nível do tanque 2 (50%), o nível do tanque 1 apresenta dois valores distintos, 65% para o primeiro ensaio (figura 4.30) e 70% para o segundo ensaio (figura 4.31). Esta diferença nos resultados deve-se ao comportamento não linear dos sensores, particularmente do sensor de caudal instalado no tanque 2.





## **Capítulo 5**

### **Conclusão**

Neste capítulo apresentam-se as conclusões gerais desta dissertação, analisando a viabilidade do sistema de controlo proposto, casos de aplicação e aspectos da sua implementação em processos industriais. Na segunda secção apresentam-se alguns pontos de trabalho futuro, a realizar sobre o processo descrito no capítulo anterior.

## 5.1 Conclusões

Nos parágrafos seguintes são apresentadas as conclusões obtidas durante a realização desta dissertação.

1. Para o caso de estudo considerado verificou-se que, apesar do ruído proveniente do sensor utilizado para medir a saída do processo, os diversos controladores projectados apresentam características robustas, assegurando o desempenho e a estabilidade do sistema.
2. Como referido anteriormente, a utilização de estruturas de controlo com componentes integrais pode, em caso de saturação dos actuadores, conduzir o sistema a regimes oscilatórios ou instáveis. A aplicação ao caso de estudo do método de *Anti-Windup* de integração condicional, revelou-se eficiente, mantendo o controlo estável mesmo em situação de saturação da bomba de circulação, no seu limite mínimo ou máximo.
3. Relativamente às linguagens de programação utilizadas, Texto Estruturado e Diagrama Funcional de Blocos, foi possível concluir que estas ferramentas possibilitam a descrição de algoritmos de complexidade considerável, constituindo por isso uma boa solução para a implementação de sistema de controlo e supervisão em ambientes industriais.
4. Durante a realização desta dissertação verificou-se alguma dificuldade em obter modelos válidos para a descrição da dinâmica do processo de dois tanques. Estas dificuldades têm origem em dinâmicas não lineares dos sensores e componentes electrónicos adjacentes, corrompendo os valores obtidos para as saídas do processo, e consequentemente, também os parâmetros dos modelos obtidos. Adicionalmente, a reduzida dimensão dos reservatórios tornou difícil a identificação dos vários regimes de funcionamento mencionados no capítulo 4. No entanto, a utilização de controladores PI com características robustas permitiu obter um desempenho satisfatório para os vários pontos de funcionamento.
5. A implementação das estruturas de controlo e de supervisão em dispositivos distintos, traduz-se na necessidade de considerar aspectos de sincronização de execução de algoritmos e de fluxo de informação entre os níveis de controlo e supervisão. Deste modo, é possível concluir que o dimensionamento destes dois níveis deve ser realizado em paralelo, de forma a cobrir todos os aspectos de complementaridade e sincronização entre os dois projectos.
6. Relativamente à estrutura em cascata adoptada para o controlo do processo de dois tanques, foi possível concluir que, para além de garantir os requisitos de desempenho em funcionamento nominal, constitui uma boa solução para a acomodação de falhas. A utilização do nível do tanque 1 como variável intermédia, permitiu reduzir o tempo de adaptação da acção de controlo às modificações introduzidas na dinâmica do processo, reduzindo também os efeitos da falha no nível do tanque 2.

Em suma, o conjunto de resultados obtidos no caso de estudo mostram a aplicabilidade do sistema de controlo tolerante a falhas proposto e evidenciam a importância dos temas abordados para o desenvolvimento de instalações industriais mais autónomas, eficazes e seguras.

## 5.2 Trabalho Futuro

Nos parágrafos seguintes são apresentados alguns pontos de trabalho futuro que se consideram importantes para o desenvolvimento das áreas de controlo tolerante a falhas e simultaneamente da área de automação.

1. Implementação dos métodos de controlo tolerante a falhas descritos no capítulo 2, utilizando as linguagens de programação para autómatos, com objectivo de desenvolver novas metodologias para a descrição de algoritmos matemáticos de maior complexidade.
2. Desenvolvimento de uma interface HMI compatível com os autómatos utilizados no caso de estudo, de forma a fornecer ao supervisor humano todas as informações sobre o estado do sistema.
3. Implementação de algoritmos que permitam guardar os dados em ficheiros, fornecendo um histórico do estado do sistema, incluindo as suas saídas, sinais de actuação, informação sobre falhas, referências, etc.
4. Estudo de protocolos e tecnologias de comunicação, por exemplo *Modbus* e *CANopen*, analisando os efeitos que as falhas na comunicação entre dispositivos introduzem no anel de controlo e de supervisão.
5. Desenvolvimento de arquitecturas e metodologias que permitam aceder remotamente ao processo considerado como caso de estudo nesta dissertação, abordando os aspectos de segurança que este tipo de projectos deve considerar.



# Bibliografia

Aström, K.J., Hagglun, T. (1995). PID Controllers: Theory, Design and Tuning, Instrument Society of America.

Aström, K.J., Hagglun, T. (1988). Automatic Tuning of PID Controllers, Instrument Society of America.

Aström, K., P. Albertos, M. Blanke, A. Isidori, W. Schaufelberger and R. Sanz (Eds.) (2001). Control of Complex Systems. Springer Verlag London.

Avizienis, A.:Infrastructure-based design of fault-tolerant systems (1998). Proceedings of the IFIP International Workshop on Dependable Computing and its Applications. DCIA 98, Johannesburg, South Africa.

Cardoso, A. (2006). Supervisão e Controlo de Sistemas Dinâmicos com Tolerância a Falha - Contribuição para uma abordagem estruturada e robusta, Universidade de Coimbra, Coimbra.

Chang, B., G. Baipai and H. Kwatny (2001). A regulator desing to adress actuator failures, Orlando, Florida, USA.

Demetriou, M. (2001). Adaptive reaorganization of switched systems with faulty actuators, Orlando, Florida, USA.

Dionísio, R., E. Moska, J. Lemos and P. Shirley (2003). Adaptive fault tolerant control with adaptive residual generation, Washington D.C., USA.

Ferreira, P. (2002). Tracking with sensor failures, Automatica, 38(9), 1621-1623.

Fliess, M., C. Join and H. Sira-Ramirez (2004). Robust residual generation for linear fault diagnosis: an algebric stting with examples, Int. Journal of Control, 77, 1223-1242.

Gil, P., A. Cardoso, J. Henriques, P. Carvalho, H. Duarte-Ramos and A. Dourado (2003), Experiments with and Adaptive Model-Based Predictive Controller Applied to a Distributed Solar Collector Field: Performance and Fault Tolerance , Almeria, Spain.

Heish, C. (2002). Performance gain margins of the two-stage LQ reliable control.Automatica, 38(11), 1985-1990.

Jiang, B., M. Staroswiecki and V. Cocquempot (2003). Active fault tolerant control for a class of nonlinear systems, Washington D.C., USA.

- Liao, F.,J. Wang and G. Yang (2002) Reliable control of non-linear systems. IEEE Transactions on Automatic Control, 45(4), 706-710.
- Liberzone D. (2003). Switched Systems, Coordinated Science Laboratory University of Illinois at Urbana-Champaign Urbana, Illinois, USA.
- Maki, M., J. Jiang and K. Hagino (2001). A stability guaranteed active fault-tolerant control system against actuator failures, Orlando Florida, USA.
- Médar, S., P. Charbonnaud and F. Nouredine (2002a). Active fault accommodation of a three tank system via switching control, Barcelona, Spain.
- Médar, S., P. Charbonnaud and F. Nouredine (2002b). Hybrid Modeling of a Switching-based Reconfiguration Strategy, Hammamet, Tunisia.
- Moreira, A., Costa, P., Santos, P. (2002). Introdução à identificação de modelos discretos para sistemas dinâmicos, Faculdade de Engenharia da Universidade do Porto.
- Niemann, H. and J.Stoustrup (2003). Reliable control using the primary and dual Youla parametrizations, Las Vegas, Nevada, USA.
- Palma, B.L. (2006). Fault Detection, Diagnosis and Fault Tolerance Approaches in Dynamic Systems Based on Black-Box Models, Portugal, Lisboa.
- Suyama, K. (2002). What is reliable control?, Barcelona, Spain.
- Theilliol, D., J. Ponsart, H. Noura and D. Sauter (2001) Sensor fault-tolerant control method based on multiple model approach, Porto, Portugal.
- Uduehi, D., Ordys, A., Grimble, M.J., Multivariable PID Controller design Using Online Generalised Predictive Control Optimisation, University of Strathclyde, Glasgow.
- Wang, Q., Ye, Z., Cai, W.,Hang, C. (2008). PID Control for Multivariable Processes, Singapura.
- Yamamoto, T., Shah, S.L. (2003). Design and experimental evaluation of a multivariable self-tuning PID controller, Hiroshima University.
- Yen, G. and L. Ho (2003). Online Multiple-Model-Based Fault Diagnosis and Accommodation, IEEE Trans. on Industrial Electronics, 50(2), 296-312.
- Zhang, Y. and J. Jiang (2001). Integrated active fault-tolerant control using IMM approach. IEEE Transactions on Aerospace and Electronics Systems, 37(4), 1221-1235.

## Anexo A.1 - Sensor de Nível

O circuito electrónico auxiliar ao qual se encontra conectado o sensor de nível do tanque 1 é ilustrado na figura 5.1. Este circuito é composto por um divisor de tensão e uma montagem seguidora, implementada com objectivo de tornar infinita a impedância de saída do circuito.

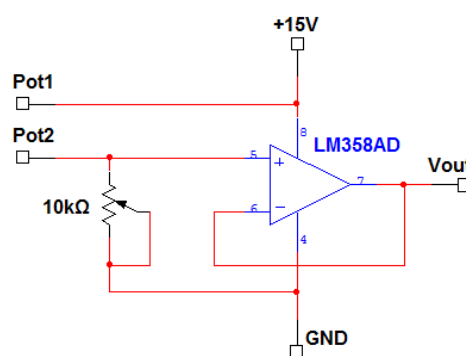


Figura 5.1: Circuito electrónico - sensor de nível

A medição é obtida através do deslocamento vertical do componente flutuador do sensor. Este deslocamento vertical é aplicado ao elemento circular e simultaneamente ao potenciómetro localizado no seu lado posterior, ilustrado na figura 5.6, tornando o seu valor resistivo proporcional ao nível. Para se obter um sinal eléctrico contínuo correspondente, os terminais do potenciómetro são conectados aos pontos Pot1 e Pot2 integrando-o desta forma no ramo superior do divisor de tensão.

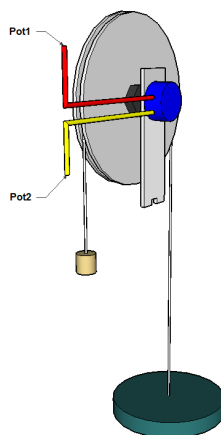


Figura 5.2: Sensor de nível





## Anexo A.2 - Sensor de Caudal

O circuito electrónico auxiliar ao qual se encontra conectado o sensor de caudal é ilustrado na figura 5.3. Este circuito é composto pelo conversor LM2907, que gera um sinal contínuo  $V_o$  em função da frequência do sinal sinusoidal  $V_{in}$ .

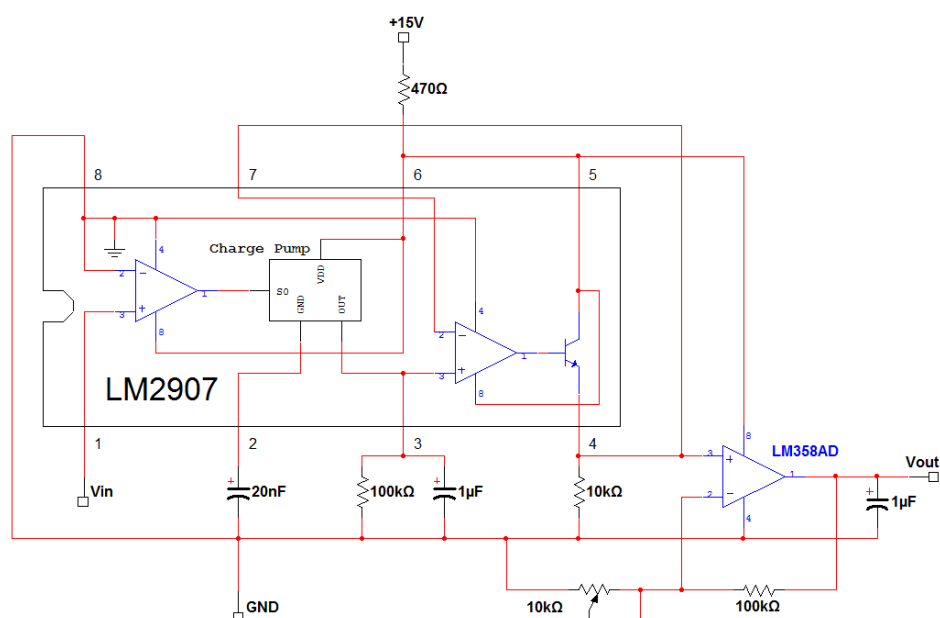


Figura 5.3: Circuito electrónico - sensor de caudal

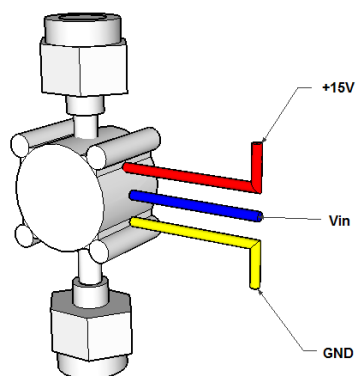


Figura 5.4: Sensor de caudal



## Anexo A.3 - Electroválvulas

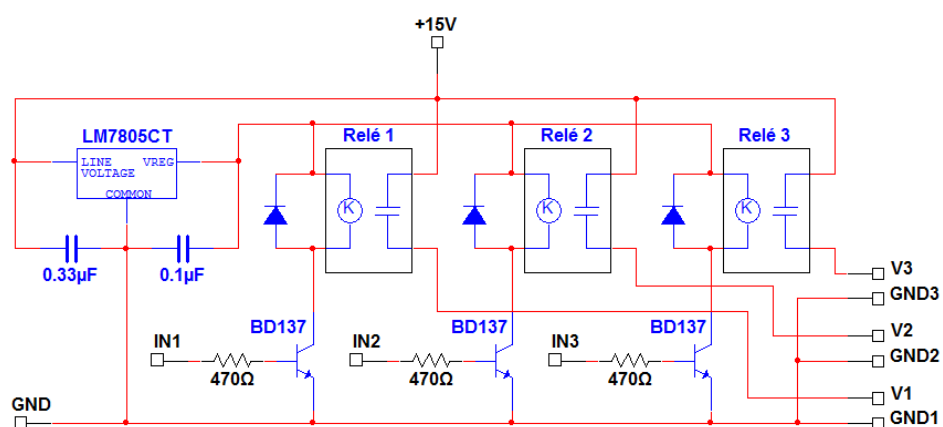


Figura 5.5: Circuito electronico - electroválvulas

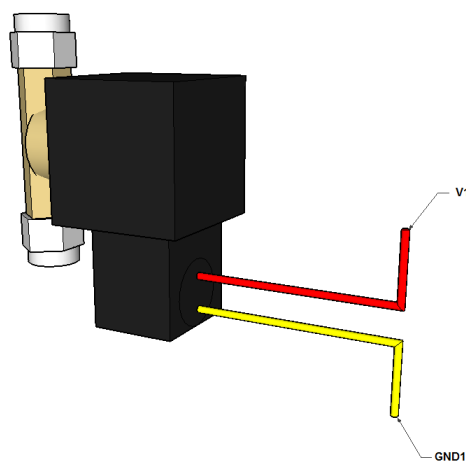


Figura 5.6: Electroválvula